FedSarah: A novel low-latency federated learning algorithm for consumer-centric personalised recommendation systems

Zhiguo Qu, Jian Ding, Rutvij H. Jhaveri, Youcef Djenouri, Xin Ning, Prayag Tiwari

***Abstract*—Data heterogeneity, insufficient scalability, and data privacy protection are the technological challenges of personalized recommendations. This study proposes a new federated learning algorithm (FedSarah) to address low scalability caused by data heterogeneity and uneven computing power in consumer-centric personalized recommendation systems while protecting data privacy of consumers. The algorithm updates the stochastic gradient estimates using a recursive framework on consumer clients. The outer loop calculates the entire gradient for updating global model, and the inner loop calculates the stochastic gradient based on the accumulated stochastic information for updating local models. To increase the stability of convergence, the inner loop modifies intrinsic parameters to change the number of training rounds and the direction of model update on consumer clients. The detailed mathematical analysis and experiments demonstrate that FedSarah has good convergence. In addition, it's shown that the algorithm can achieve a performance improvement of nearly 5% in terms of accuracy compared to the traditional FedAvg and FedProx algorithms under the condition of heterogeneous data. Furthermore, under the condition of effective privacy protection on consumers' data, the new algorithm can significantly lessen the impact of data heterogeneity on the real-time service of consumer-centric personalized recommendation systems with low communication latency.**

***Index Terms*—Federated Learning, Consumer-centric Personalized Recommendation System, Data Heterogeneity, Uneven Consumer Computing Power.**

I. Introduction

MANUFACTURING and retailing of electronic goods directly to consumers is referred to as the consumer electronics industry. This industry includes a vast variety of electronic goods, such as computers, televisions, audio equipment, smartphones, etc. With the prosperity of the Internet and the popularization of wireless networks, both the number of consumer electronics devices and the amount of data that they produced are expanding at an exceedingly rapid rate. A new study predicts that by 2025, there will be 2,873.1 million users in the consumer electronics industry, with an average revenue per user (ARPU) of \$317.1 billion [1]. Every device can be used to create and share data online, which has facilitated the development of consumer electronics. The consumer electronics sector is gradually implementing personalized recommendation systems to improve the consumer experience as a key technology that continues to advance [2]. In order to assist consumers in finding the most helpful products or services at the ideal moment and in the most suitable method, consumer-centric personalized recommendation systems have received a lot of attention. These systems aim to address various real-world issues with information overload. Examples include e-commerce, healthcare, and news push.

Personalized recommendation systems aim to provide tailored product or service suggestions to users, with two main approaches: collaborative filtering and content-based recommendations. Content-based recommendation identifies user preferences and product features, presenting a collection of products that align with users' interests. It utilizes historical data to understand past consumer preferences, analyzes product features, and suggests comparable items [3]. Collaborative filtering focuses on personalization by recommending products based on the preferences of users with similar characteristics without requiring detailed product knowledge. Its effectiveness relies on the degree of similarity between consumer groups, establishing connections between potential and existing consumers to offer personalized recommendations based on shared preferences [4].

Consumer-centric personalized recommendation systems that are currently available have produced more notable accuracy outcomes, but one of their drawbacks is that they must send all the consumers' private data to a central server for storage, which could result in the issue of privacy leakage. The importance of overcoming the heterogeneity problem of individual consumers due to different storage, computing powers, and communication capabilities has greatly increased as a result of the gradual improvement of data privacy laws, the enhancement of regulations (e.g. GDPR), and the increased awareness of consumers regarding the protection of their respective private data in recent years. In addition,

This work was supported by the National Natural Science Foundation of China (No. 61373131, 62071240), PAPD and CICAEET funds. *Corresponding Authors: Rutvij H. Jhaveri, Zhiguo Qu, Prayag Tiwari*

Zhiguo Qu is with the Jiangsu Collaborative Innovation Center of Atmospheric Environment, the Equipment Technology and the Engineering Research Center of Digital Forensics, Ministry of Education, and the School of Computer Science, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: qzghhh@126.com).

Jian Ding is with the School of Computer Science, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: dj2213352201@126.com).

Rutvij H. Jhaveri is with the Department of Computer Science and Engineering, School of Technology, Pandit Deendayal Energy University, India (E-mail: rutvij.jhaveri@sot.pdpu.ac.in)

Youcef Djenouri is with the NORCE Norwegian Research Center Oslo, Norway University of South-Eastern Norway, Konsberg, Norway(E-mail: yodj@norceresearch.no)

Xin Ning is with the Laboratory of Artificial Neural Networks and High Speed Circuits, Institute of Semiconductors, Chinese Academy of Sciences. (E-mail: ningxin@semi.ac.cn)

Prayag Tiwari is with the Department of Computer Science, Aalto University, Finland (Email: prayag.tiwari@ieee.org)

the majority of intelligence and decision-making processes in typical Internet of Things (IoT) designs are centralized in the cloud. However, sending all the data to the cloud for processing and analysis may result in latency, bandwidth challenges, and privacy concerns as the number of IoT devices keeps growing and the amount of data is expanding drastically as well. Edge intelligence is progressively emerging as a hub for IoT research and applications to address these issues [5]. In these environments, intelligence is mostly realized in central controllers, while IoT devices have nearly no room to communicate, share knowledge, or draw on the knowledge of their peer devices. Dehury et al. consequently proposed a clustering and cohesion strategy to create one or more groups [6]. It is clear that federated learning models can aid in decentralizing the training of personalized recommendation systems that are usually centered on consumers, solving these diverse issues, and creating clusters to increase the effectiveness of device collaboration.

In spite of the fact that currently available federated learning models may train the global model without requiring consumers' data to leave their devices, consumers are still vulnerable to performance degradation and increased communication latency because of the heterogeneity of the data used in federated learning. In this case, if consumers must wait for a considerable period of time to receive high-caliber personalized recommendations, these issues will negatively impact consumers' experiences and decrease the acceptance of personalized recommendation systems [7]. Therefore, federated learning-based personalized recommendation systems must increase the speed and accuracy of their recommendations.

In this study, FedSarah is proposed as a novel federated learning algorithm that attempts to overcome the issues of data heterogeneity and low scalability in consumer-centric personalized recommendation systems while achieving consumers' privacy protection. In consumer-centric personalized recommendation systems, the issue of inconsistent data distribution across consumers needs to be examined first. It's because federated learning is made more difficult by the heterogeneity of the data because various consumers' data may have distinct traits and preferences. Second, it's necessary to strike a balance between the efficiency of their communication and the consideration of consumers with varying computing powers. In order to achieve this goal, we must create an algorithm that can adjust to consumers with different computing powers and lessen the influence of computing power variances on the results during global model aggregation since consumers may have differences in device performance and communication efficiency. To solve these issues, the FedSarah algorithm uses the parameter $v_{n,s}^{(t-1)}$ to indicate the update direction of the global model, while $\nabla f_u(w_{n,s}^t) - \nabla f_u(w_{n,s}^{t-1})$, the local gradient of the loss function, indicates the update direction of the consumer model. And then, we are able to correct consumer models that diverge from the direction of global model updates, resulting in more precise global model aggregation. Additionally, the FedSarah algorithm provides a parameter γ to set the accuracy of model updates at the consumer to deal with some consumers with limited computing power. After that, it can further enhance the system's scalability and reduce communication latency by lowering the number of local update rounds for the consumers with insufficient computing power. In conclusion, the FedSarah algorithm offers a novel approach to federated learning in personalized recommendation systems. It can successfully mitigate the effects of data heterogeneity and computing power differences on global model aggregation while ensuring the security of consumer data.

The main works and contributions of this paper are presented as follows:

- For consumer-centric personalized recommendation systems, this study proposes FedSarah, a unique low-latency federated learning algorithm. The algorithm can effectively mitigate the adverse effects of consumer data heterogeneity and computational power differences on the operation of the recommendation system while also protecting consumer privacy. It does this by self-optimizing and adding correction terms to minimize data interference from other consumers.
- The FedSarah algorithm's convergence is proved by rigorous mathematical derivation in this study. It ensures that the FedSarah algorithm can converge to a stable state after a predetermined number of training rounds.
- The experiments have shown that the FedSarah algorithm, used in consumer-centric personalized recommendation systems, has more accuracy and balance than the baseline algorithms. In this study, various datasets and segmentation methods are employed to analyze FedSarah's performance from various angles. The findings demonstrate that FedSarah is capable of improving algorithmic scalability while minimizing the detrimental effects of data heterogeneity and computing power disparities on system performance and reducing communication latency.

The rest of the paper is organized as follows: Section 2 presents the related work. Section 3 details the design idea and implementation of FedSarah and provides a complete mathematical derivation for proving its convergence. Section 4 describes the experimental results of FedSarah on multiple datasets and also gives an illustration. Finally, Section 5 gives its conclusion.

II. Related work

Federated learning is a concept that was developed by Google in 2016 and allows for data sharing, collaborative modeling, and legal compliance. McMahan, who created the FedAvg framework, demonstrated the viability of federated learning [7]. Applications of federated learning in other domains have evolved since Google first suggested using it on Android devices to anticipate users' Gboard inputs. In 2017, without directly sharing patient-level data, Kim et al. [8] presented a secure data coordination and joint calculation approach based on the alternating direction multiplier (ADMM) to jointly derive phenotypes from various hospitals. In 2018, in order to locate similar instances from one hospital to another without disclosing patients' private information, Lee et al. [9] proposed a new technique to learn context-specific hash codes to represent patients across different institutions. In the same year, in order to train general-purpose classifiers for the usage of intelligent gadgets to predict human behavior, Sozinov et al. [10] proposed a new federated learning algorithm. Also in the same year, Zhao et al. [11] presented that weights could be expressed in terms of the separation between the global distribution and the category distribution on each device. As a result, they proposed a method for reducing the weight dispersion issue brought on by non-independent, identically distributed data by producing a limited subset of data that is shared globally among all edge devices. In 2019, to forecast the energy demand of EV networks, Saputra et al. [12] presented a collaborative strategy based on energy demand learning. In the same year, in order to lessen the losses brought on by credit card theft for banks and consumers, Yang et al. [13] proposed a federated learning algorithm for theft detection (FFD) by establishing a methodology for training fraud detection models adopting behavioral characteristics and federated learning. Also in the same year, Mohri et al. [14] drew attention to how prior research had overlooked the significance of fairness, which had resulted in bias against various clients. Therefore, a new framework for agnostic federation learning was put forth. It treats the target distribution as a convex combination of various source distributions and then converts it into a min-max problem to optimize performance on the convex combination of distributions that has the biggest influence on the model. In 2020, in order to allow cloud servers to combine diverse knowledge from robots all over the world without worrying about privacy and security concerns, Liu et al. [15] proposed a federated learning-based data fusion method for robot imitation learning. This method offers guidance models for robots submitting service requests. In the same year, FedProx was proposed by Li et al. [16]. To reduce the amount of local variance, FedProx adds an additional proximal term to the local aim. In 2021, a general framework called Ditto that takes into account fairness and robustness in federated learning systems was proposed by Li et al. [17]. In 2022, Oh et al. [18] proposed to construct a good global model by using individualized federated learning. Their analysis of the entire network into subjects associated with universality and heads related to personalization helped to clarify the causes of the decline in federated learning personalization performance. In 2023, the PDP-PFL algorithm was developed by Qu et al. [19]. It is a personalized federated learning approach with privacy preservation that optimizes both information sharing and data privacy for smart vehicle networking through hierarchical privacy preservation and local parameter preservation.

So far, researchers have used federated learning in numerous fields as federated learning technology has gradually matured. For example, in 2022, to distinguish between federated learning (FL) with structured data and structured federated learning (structured FL), Fu et al. [20] proposed a novel classification approach for problems in federated graph machine learning (FGML). The survey also presents commonly employed open graph datasets and platforms in FGML research, provides an overview of the practical applications of FGML across diverse domains, and underscores the existing research limitations while pointing out promising avenues for future investigation. In 2023, Li et al. [21] presented formal definitions in accordance with their discussion of the transition from conventional machine learning to domain adaptation and domain generalization. Furthermore, the most modern methods are categorized. By storing the consumer's personal information locally and using the intermediate parameters of the server and clients for collaborative optimization, federated learning, a type of privacy-protecting distributed machine learning framework, can ensure the prediction performance of machine learning models while simultaneously protecting consumers' personal privacy [7]. Among them, recommendation systems based on federated learning have also drawn a lot of attention from researchers because it makes sense for recommendation algorithms to move from a centralized learning framework to a federated learning paradigm in order to satisfy the need for consumers' privacy protection.

Recommendation systems play a significant role in reducing the issue of information overload by removing items that may be of interest to consumers and enhancing their consumption experience. To increase the recommendation accuracy index in the early development stages of recommendation systems, people typically employed more complicated models or fused more data. As consumers' expectations of the experience of recommendation systems gradually increase, researchers have begun to research more intelligent and user-friendly recommendation systems with privacy protection. For example, in 2017, Guo et al. [22] proposed the DeepFM model, which integrates deep learning for feature learning and the capabilities of a deconstruction machine for recommendations in a new neural network architecture. In 2018, Liang et al. [23] presented a generative model with multinomial likelihood using Bayesian inference for parameter estimation and extended the variational autoencoder (VAE) to collaborative filtering for implicit feedback. In 2019, in order to jointly capture interactions and opinions in user-item graphs, Fan et al. [24] introduced a novel graph neural network framework (GraphRec) for social recommendations. In 2020, Beigi et al. [25] proposed a novel recommendation with attribute protection (RAP) model, which concurrently suggests pertinent items and guards against attacks via private attribute inference. In recent years, federated learning has been used to build personalized recommendation systems due to the advantages of decentralization and privacy protection. For example, in 2020, by merging federated learning and personalized recommendations, Muhammad et al. [26] proposed FedFast for accelerating distributed learning to increase the precision of personalized suggestions. In the same year, following the recently developed Federated Collaborative Filtering (FCF) technique for implicit feedback item recommendation, Lin et al. [27] proposed the FedRec framework, a novel and all-encompassing framework that combines a number of fundamental and sophisticated recommendation models based on factorization in a batch- and stochastic-based manner for rating prediction with explicit feedback. In 2021, a multi-armed bandit technique was developed by Khan et al. [28] to solve the model payload issue by choosing a component of the global model and sending it to all the consumers. In 2022, a privacy-preserving GNN-based joint recommendation method was put forth by Wu et al. [29] that uses higher-order user-item interaction data to leverage collaboratively trained GNN models from dispersed user data. In conclusion, federated learning for personalized recommendation has many achievements and has become an important research branch of federated learning. So far, most studies have focused on finding ways to mitigate the drawbacks of data heterogeneity in federated recommendation systems. Therefore, in the new algorithm proposed in this paper, solving the problem of poor and delayed recommendations caused by data heterogeneity from the perspective of consumers is our core starting point.

III. FedSarah's processing steps and its proof of convergence

In this section, the fundamental challenges dealt with by federated learning in a consumer-centric personalized recommendation system are briefly described, then the specific processing steps of FedSarah are presented in detail, as well as a convergence proof given for the new algorithm.

In a typical recommendation system (RS), consumers are frequently heterogeneous, with varying tastes for various products. Although most existing recommendation systems benefit from similar features shared with different consumer groups, most current federated learning based recommendation systems have not conducted in-depth research and exploration on this characteristic. As a result, in a consumer-centric personalized recommendation system, the main goal of federated learning should be to achieve local models that can effectively handle large amounts of data collected from distributed consumers, as well as a central server that can cooperate with overall learning goals. Let's formalize this purpose into mathematical terms as follows:

$$\min_{w \in R^d} F(w) = \sum_{n=1}^{N} \frac{D_n}{D} F_n(w) \tag{1}$$

$$F_n(w) = \frac{1}{D_n} \sum_{i \in D_n} f_i(w) \tag{2}$$

Here, N represents the total number of consumers. And for each n-th consumer, it has a local data set D_n. D_n denotes the amount of the data volume on the n-th consumer and D is the sum of the data volumes of all consumers, satisfying with $\sum_{n=1}^{N} \frac{D_n}{D} = 1$. The loss function for the n-th consumer is given by Equation 2, while Equation 1 represents the global loss minimization issue.

If we would like to increase the learning efficiency of consumer clients in an environment of data heterogeneity and uneven computing power in a consumer-centric personalized recommendation system, the key lies in how to improve the flexibility of consumer clients and optimize the recommendation efficiency. Thus, in order to mitigate the negative effects of data heterogeneity, this paper introduces a novel federated learning algorithm named FedSarah. By including a correction term in the local function, it lessens the impact of equally and non-independently distributed data, increasing the precision of recommendations. In addition, a parameter γ is used to enable the client to dynamically adjust the number of local computation rounds in accordance with its own circumstances, reducing the impact of the communication delay brought by the different computing power on the global model aggregation, thus reducing communication delays and improving the accuracy of the recommendation. This is carried out in an effort to solve the issue of global model aggregation caused by the different computing power of each consumer client, thus improving the speed of recommendation. It is further proved that FedSarah can converge sublinearly in a single inner loop under the general convexity assumption and linearly under the strong convexity assumption. The specific processing steps of FedSarah are presented in detail as follows, as well as its convergence proof.

A. The processing steps of FedSarah

The optimization strategy of adding additional gradient correction terms to the local objective function, as recommended by the distributed methods DANE [30] and AIDE [31], is incorporated into FedSarah. Modifications to some of these steps allow the FedSarah algorithm to make significant empirical improvements in the face of the issue of data heterogeneity for the consumer-centric personalized recommendation system. In the new algorithm, consumer-client representations are kept on the individual consumers' local devices to reduce the possibility of privacy leakage. Since a single consumer's data is insufficient to train a customized recommendation model, many consumers are needed to jointly train the model. To reduce the communication and computation overheads at the consumer clients and enhance the speed and accuracy of personalized recommendations, we decompose the personalized recommendation model into a large personalized recommendation model maintained by the server and numerous lightweight consumer models shared by the server and consumer clients. The four main steps that make up model updating are system initialization, local calculation, central aggregation, and model update. Figure 1 displays the flowchart of the new algorithm.

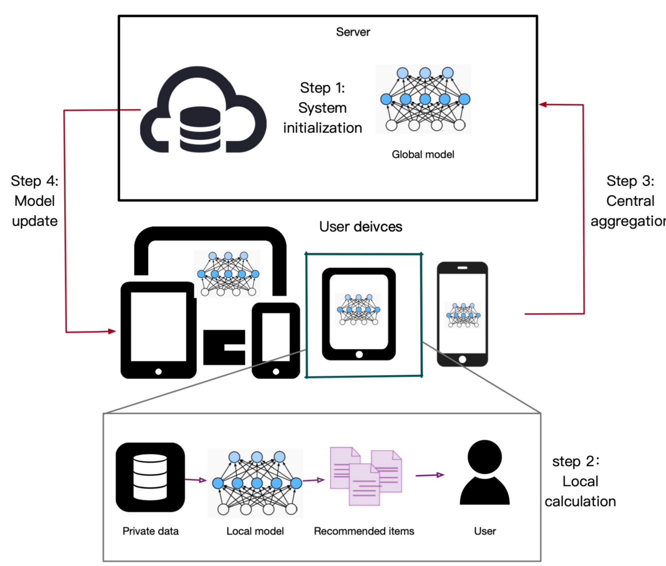

Fig. 1. The flowchart of FedSarah

Step 1: System initialization. A central server initiates the training task and dynamically chooses a random sample of consumers to take part in the training of the personalized recommendation model. The consumer client's selection of whether or not to join in the cooperative modeling is mainly based on their demands. After all participating consumer clients are validated, the central server transmits initial parameters to each consumer client, including the learning rate η, the parameter γ, the local update cap m, and the initialized model parameter w_0.

Step 2: Local calculation. After every consumer client has received the initial parameters from the server, the consumer client does a local computation based on the local data. The previous gradient information will be learned by means of the correction term $v_{n,s}^{(t-1)}$, and the present gradient information $\nabla f_u(w_{n,s}^t) - \nabla f_u(w_{n,s}^{t-1})$ will be merged to update the gradient. After each update, the consumer client checks whether the desired precision is attained by means of the parameter γ and chooses whether or not to proceed with the next round of local update operations. Here, m is used to regulate the maximum number of local updates. After the computation is done, the locally determined gradient is desensitized and communicated to the server to support the global model update. The effect of the correction term is displayed in Figure 2.

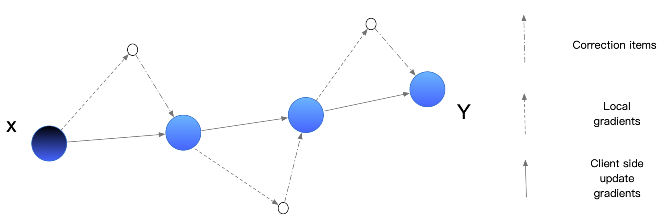

Fig. 2. Diagram of correction term

Step 3: Central aggregation. Following receipt of the computation results from each participating consumer client by the server, the results are weighted and aggregated based on the volume of data for each client.

Step 4: Model update. The global model is updated separately from the consumer recommendation model. Based on the aggregated results, the server updates the global model, which is then transmitted to the consumer clients. The consumer client receives and updates its local model. Then, the performance of the new model is evaluated before proceeding to the subsequent step of the local computation. Once the performance has been attained, training will be halted and the modeling will be accomplished.

In the pseudocode of FedSarah (Algorithm 1), a $v_{n,s}^{(t)}$ (in line 9) and γ, m (in line 1) shall be noticed. $v_{n,s}^{(t)}$ is a very popular method in the field of optimisation, and it's called as the correction term. While the parameters γ, m are used to regulate local accuracy requirements and a maximum number of iterations for the consumer client, respectively. The meanings of the remaining parameters are shown in Table I.

Algorithm 1 FedSarah

Input: η, γ, m, w_0
for $s = 1, ..., T$ **do**
  for $n = 1, ..., N$ **do**
    $w_{n,s}^{(0)} = w_{s-1}$
    $v_{n,s}^{(0)} = \nabla F_n(w_{n,s}^{(0)})$
    $w_{n,s}^{(1)} = w_{n,s}^{(0)} - \eta v_{n,s}^{(0)}$
    t = 1
    while $||v_{n,s}^{(t-1)}||^2 > \gamma ||v_{n,s}^{(0)}||^2 \;\; and \;\; t < m$ **do**
      Sample u uniformly at random from $[D_n]$
      $v_{n,s}^{(t)} = \nabla f_u(w_{n,s}^{(t)}) - \nabla f_u(w_{n,s}^{(t-1)}) + v_{n,s}^{(t-1)}$
      $w_{n,s}^{(t+1)} = w_{n,s}^{(t)} - \eta v_{n,s}^{(t)}$
      t = t + 1
    end while
    set $w_n^{(s)} = w_{n,s}^{(t)}$
  end for
  $w_s = \sum_{n=1}^{N} \frac{D_n}{D} w_n^{(s)}$
end for

TABLE I
SYMBOLIC ANNOTATION

s, n, t	Index of the global number of iterations, index of the consumer client, index of the number of consumer client iterations
N, m	Total number of consumer clients, maximum number of consumer client iterations
D, D_m	Sum of the data volume of all consumer clients, the *n*-th consumer client data
$w_m^{(s)}, w_s$	Training result for the *n*-th consumer client in round s, the global model after the *s*-th round of aggregation
$w_{n,s}^{(t)}$	The *t*-th local training result for the *n*-th consumer client in *s*-th round
$v_{n,s}^{(t)}$	Update direction of the *n*-th consumer client for the *t*-th local training in the *s*-th round

Regarding $v_{n,s}^{(t)}$, its explanation is given as follows. The impact of data heterogeneity on a consumer-centric personalized recommendation system is mainly due to the fact that the data of each consumer client is unindependent and unidentically distributed. This makes the update direction at the local update of each consumer side inconsistent with or even opposite to the update direction of the system's global model, which leads to slower convergence and affects the accuracy of personalized recommendations. To solve this potential issue, this paper designs a parameter $v_{n,s}^{(t)}$ based on the common optimization method correction term in the field of optimization. As it can be observed in the line 4 and 9 of FedSarah's pseudocode, $v_{n,s}^{(0)}$ represents the global initial model. Each successive update to the local model will regulate the direction in which the model updates to connect with the global model through $v_{n,s}^{(t)}$. This improves the precision of recommendations by reducing the effect of data heterogeneity on global model aggregation.

Regarding the parameters γ, m, the explanations are given as follows. The influence of uneven computing power on consumer-centric personalized recommendation system is mainly due to the different chips in consumer clients and the resulting communication latency problem. Therefore, the maximum number of computation rounds for multiple consumer clients at the same time should also be adjusted correctly according to their different local computing powers. In FedSarah, we first specify the maximum number of rounds to be updated locally by using the parameter m for all the consumer clients. After mastering the local computing powers of consumer clients, it can use the parameter γ to control the number of rounds to be updated locally by consumer clients. As a result, it effectively increases the speed of personalized recommendations while maintaining a high level of accuracy and decreasing the communication latency issue caused by uneven system computing power.

B. Proof of convergence of FedSarah's Algorithm

In this subsection, it will be proven that FedSarah's stochastic steps converge linearly within the inner loop. In addition to presenting the most popular optimization algorithms and exploring alternative strategies for optimizing gradient descent, Ruder et al. [32] provided the underlying theory of gradient descent optimization algorithms. Zinkevich M et al. [33] provided an explanation of convex function optimization problems to establish a generally consistent extension of the infinitesimal gradient ascent algorithm. In order to continue analyzing the proposed algorithm, we make the following assumptions based on the theoretical foundations of the two papers mentioned above.

Assumption 1 (L-smooth). Each f_i is L-smooth, $R^d \rightarrow R$, $i \in [D_n]$, in other words, there is always a constant L>0, so that

$$L||w - w'|| \geq ||\nabla f_i(w) - \nabla f_i(w')||, \forall w, w' \in R^d. \quad (3)$$

This signifies that $F_n(w) = \frac{1}{D_n}\sum_{i \in D_n} f_i(w)$ is also L-smooth.

Assumption 2 (μ-strongly convex). The function $F_n(w)$ is μ-strongly convex, $R^d \rightarrow R$, in other words, there is always a constant $\mu > 0$, so that for any $w, w' \in R^d$,

$$F_n(w) \geq F_n(w') + \nabla F_n(w')^T(w - w') + \frac{\mu}{2}||w - w'||^2. \quad (4)$$

Because the analysis requires, a stronger μ-strongly convex will also be imposed on Assumption 1. Note that Assumption 3 integrates Assumption 2, while the reverse is not true.

Assumption 3. Each function f_i is strongly convex when $\mu > 0$, $R^d \rightarrow R$, $i \in [D_n]$.

Set w^* be the best solution of 2 under Assumption 2, F_n strongly convex signifies that

$$[F_n(w) - F_n(w^*)] \leq \frac{||\nabla F_n(w)||^2}{2\mu}, \forall w \in R^d. \quad (5)$$

It is essential to remember that, for the future use, the number of conditions is defined as $\kappa \stackrel{def}{=} \frac{L}{\mu}$ for substantially convex machine learning functions as Equation 2. In addition to this, it is vital to note that Assumption 2 and Assumption 3 encompass a good number of different issues, such as the l_2-regularised empirical risk minimisation problem with convex losses.

Assumption 4. Each function f_i is convex, $R^d \rightarrow R, i \in [D_n]$, in other words,

$$f_i(w) \geq f_i(w') + \nabla f_i(w')^T(w - w'), \forall i \in [D_n]. \quad (6)$$

It is worth noting that, if Assumption 3 is true, Assumption 4 will be also true. However, if Assumption 2 is true, then Assumption 4 will be false. In the analysis of this section, either Assumption 4 must hold independently, Assumption 2 and Assumption 4 must hold together, or Assumption 3 must hold independently. Here, Assumption 1 will be set to true by default.

The iterative complexity analysis is initially done, and its major objective is to reduce the total number of stochastic gradient evaluations. This requires the guarantee: $||\nabla F_n(w_\tau)||^2 \leq \epsilon$. In this case, w_τ will be regarded as an ϵ-exact solution. Similar to the normal practice of stochastic gradient algorithms, for obtaining a maximum bound on the squared norm of a forecast gradient, FedSarah must first establish a bound on the number of iterations. In other words,

$$E[||\nabla F_n(w_\tau)||^2] \leq \epsilon. \quad (7)$$

In the inner loop, it will be demonstrated that FedSarah's stochastic steps converge linearly. Two linear convergence outcomes based on two distinct μ-strongly assumptions are provided below:

Theorem 1. Assume that Assumptions 1, 2 and 4 are true, and think about the $v_{n,s}^{(t)}$ specified in line 9 of FedSarah's pseudocode with $\eta < \frac{2}{L}$. Then, for $\forall t \geq 1$,

$$\begin{aligned} E[||v_{n,s}^{(t)}||^2] &\leq E[||v_{n,s}^{(t-1)}||^2][1 - \mu^2\eta^2(\frac{2}{\eta L} - 1)] \\ &\leq [1 - (\frac{2}{\eta L} - 1)\mu^2\eta^2]^t E[||\nabla F_n(w_0)||^2]. \end{aligned} \quad (8)$$

This conclusion indicates that by selecting $\eta = O(\frac{1}{L})$, it is observed that $||v_{n,s}^{(t)}||^2$ converges linearly at a rate of $(1 - \frac{1}{\kappa^2})$. The next show that in the strongly convex assumption, a greater rate of convergence is attained.

Theorem 2. Assume that Assumptions 1 and 3 are true, and think about $v_{n,s}^{(t)}$, as specified in the line 9 of FedSarah's pseudocode with $\eta \leq 2(\mu + L)$. Then, for $\forall t \geq 1$,

$$\begin{aligned} E[||v_{n,s}^{(t)}||^2] &\leq (1 - \frac{2\mu L\eta}{\mu + L})E[||v_{n,s}^{(t-1)}||^2] \\ &\leq (1 - \frac{2\mu L\eta}{\mu + L})^t E[||\nabla F_n(w_0)||^2]. \end{aligned} \quad (9)$$

Here again, by setting $\eta = O(\frac{1}{L})$, the linear convergence rate may be increased to approach $(1 - \frac{1}{\kappa})$. There is a significant improvement in the rate of convergence compared to the prior one.

Having proved previously that the inner loop of the FedSarah algorithm converges linearly, the general convergence rate results for FedSarah will be derived as follows. Before deriving, two important Lemmas are initially provided as theoretical underpinnings. It is then proved independently that a single external iteration has a sublinear convergence rate when it's applied to a general convex function. Also, when it's applied to a significantly convex function, the convergence rate of the procedure with numerous external iterations is linear.

Lemma 1. Assume that Assumption 1 is true, think about FedSarah, it's easy to deduce that

$$E[||\nabla F_n(w_{n,s}^{(t)})||^2] \leq \frac{2}{\eta} E[F_n(w_{n,s}^{(0)}) - F_n(w^*)] + \sum_{t=0}^{m} E[||\nabla F_n(w_{n,s}^{(t)}) - v_{n,s}^{(t)}||^2] - (1 - L\eta) \sum_{t=0}^{m} E[||v_{n,s}^{(t)}||^2]. \tag{10}$$

Lemma 2. Assume that Assumption 1 is true, and think about $v_{n,s}^{(t)}$, as specified in the line 9 of FedSarah's pseudocode. Then, for any $t \geq 1$,

$$E[||\nabla F_n(w_{n,s}^{(t)}) - v_{n,s}^{(t)}||^2] = \sum_{i=1}^{t} E[||v_{n,s}^{(i)} - v_{n,s}^{(i-1)}||^2] - \sum_{i=1}^{t} E[||\nabla F_n(w_{n,s}^{(i)}) - \nabla F_n(w_{n,s}^{(i-1)})||^2]. \tag{11}$$

The convergence rate for the general convex case is firstly proved as follows.

According to Lemma 2, an upper bound on $E[||\nabla F_n(w_{n,s}^{(t)}) - v_{n,s}^{(t)}||^2]$ can be obtained for a convex function $f_i, i \in [D_n]$.

Lemma 3. Assume that Assumptions 1 and 4 are true, and think about $v_{n,s}^{(t)}$, as specified in the line 9 of FedSarah's pseudocode with $\eta < \frac{2}{L}$. Then, for $\forall t \geq 1$,

$$E[||\nabla F_n(w_{n,s}^{(t)})||^2] \leq \frac{\eta L}{2 - \eta L} [E[||v_{n,s}^{(0)}||^2] - E[||v_{n,s}^{(t)}||^2]] \leq \frac{\eta L}{2 - \eta L} E[||v_{n,s}^{(0)}||^2]. \tag{12}$$

A core theorem can be proved by using the aforementioned lemmas as follows.

Theorem 3. Assume that Assumptions 1 and 4 are true, and think about FedSarah with $\eta < \frac{1}{L}$. Then, for $\forall s \geq 1$,

$$E[||\nabla F_n(w_n^{(s)})||^2] \leq \frac{2}{\eta(m+1)} E[F_n(w_n^{(s-1)}) - F_n(w^*)] + \frac{\eta L}{2 - \eta L} E[||\nabla F_n(w_n^{(s-1)})||^2]. \tag{13}$$

The following convergence results can be obtained with $\eta = \sqrt{\frac{2}{L(m+1)}}, \sqrt{\frac{2}{L(m+1)}} \leq \frac{1}{L}$,

$$E[||\nabla F_n(w_n^{(s)})||^2] \leq \frac{\sqrt{2L}}{\sqrt{m+1}} E[F_n(w_n^{(s-1)}) - F_n(w^*) + ||\nabla F_n(w_n^{(s-1)})||^2]. \tag{14}$$

This result demonstrates the sublinear convergence rate of FedSarah within a single inner loop under the assumption of a generic convex function.

The linear convergence rate of FedSarah in the strongly convex situation is then derived. According to Theorem 2., if $s \geq 1$, paired with Equation 5, then

$$\begin{aligned} E[||\nabla F_n(w_n^{(s)})||^2] &\leq \frac{2}{\eta(m+1)} E[F_n(w_n^{(s-1)}) - F_n(w^*)] + \frac{\eta L}{2 - \eta L} E[||\nabla F_n(w_n^{(s-1)})||^2] \\ &\leq (\frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2 - \eta L}) E[||\nabla F_n(w_n^{(s-1)})||^2]. \end{aligned} \tag{15}$$

It's equivalent to

$$E[||\nabla F_n(w_n^{(s)})||^2] \leq \sigma_m E[||\nabla F_n(w_n^{(s-1)})||^2]. \tag{16}$$

Definition: $\sigma_m \overset{def}{=} \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L}$. By choosing appropriate η and m such that $\sigma_m < 1$, the following convergence result can be derived by combining Equation 16.

Theorem 4. Assume that Assumptions 1, 2 and 4 are true, by setting m and η, so that $\sigma_m \overset{def}{=} \frac{\eta L}{2-\eta L} + \frac{1}{\mu\eta(m+1)} < 1$. Thus, it is possible to obtain

$$E[||\nabla F_n(w_n^{(s)})||^2] \leq (\sigma_m)^s E[||\nabla F_n(w_n^{(0)})||^2]. \tag{17}$$

As a result, the conclusion can be obtained that FedSarah converges linearly in the strongly convex situation.

IV. Experiments

In this section, consumer-centric personalized recommendation scenarios will be simulated using synthetic datasets and some realistic datasets. The recommendation effect of the FedSarah algorithm in consumer-centric personalized recommendation scenarios will then be demonstrated from multiple perspectives, respectively. For clarity, the details of the experiments will first be presented. Then, the optimization of FedSarah will be proved by comparing its performance with that of FedAvg and FedProx. And finally, the tolerance of FedSarah with regard to the uneven computing power will be shown by adjusting the parameter γ.

A. Experimental initial setup

In our research, we conduct a thorough evaluation of the FedSarah algorithm within the framework of a consumer-centric personalized recommendation system. Our evaluation encompasses a diverse set of datasets, including the Synthetic dataset proposed by Li et al. [16], which features both non-independently and identically distributed data and independently and identically distributed data. Additionally, we extend our performance evaluation to include several benchmark datasets, namely, the Fashion_MNIST dataset, the Movielens-100K dataset [34], and the ASSIST dataset [35]. These datasets are deliberately chosen to provide a comprehensive assessment of the FedSarah algorithm's efficacy and adaptability across various recommendation scenarios. The Fashion_MNIST dataset encompasses a diverse collection of 70,000 frontal images showcasing various clothing items across 10 distinct categories. The MovieLens-100K dataset meticulously records a sum of 100,000 ratings given to 1,682 movies by 943 users. Importantly, each user contributed ratings for at least 20 movies, providing a substantial and rich dataset for our experiments. The ASSIST dataset (2009-2010 "Non-skill builder") captures the Learning Logs of the Non-skill Maths Tutoring Programme conducted during the 2009-2010 academic year. In the experiments, the datasets are pooled and then randomly divided, with three-quarters being used for training and one-quarter being utilized for testing.

For the Fashion_MNIST dataset, 100 consumer clients are simulated as part of the simulation experiments, and the local model is a multi-class logistic regression model. To better test the effectiveness of recommendations, each consumer client only has two samples, and samples are distributed according to the Power-law rule. The local model input is 28*28 grayscale images, and the output is the class label information. For all synthetic datasets, 30 consumer clients are simulated, and their sample distribution likewise follows the Power-law rule. In the MovieLens100K dataset, we divided the data into multiple consumer clients based on the country region shown by the user's zipcode (e.g., 1xxxx-Delaware, New York, Pennsylvania) and removed the consumer clients with fewer than 5 users. After that, we have the MovieLens-100K dataset with more than 96K records from 925 users from 10 consumer clients. Users rated 1676 products (i.e., movies) belonging to 19 categories on a five-point scale. Similarly, in the ASSIST dataset, we divided the data into multiple consumer clients based on the teacher ID. In addition, we filtered out consumers with fewer than five students and fewer than five average records to ensure that the algorithm has enough data for training. In the end, we obtained more than 300,000 records from 3,477 students across 59 consumer clients. More than 17,000 questions were answered on a two-point scale for 122 concepts. The data distributions for MovieLens-100K and ASSIST are shown in Table II.

TABLE II
MOVIELENS-100K, ASSIST DATA STATISTICS

Statistics	MovieLens-100K	ASSIST
consumer clients	10	59
records	96,538	327,058
users	925	3,477
items	1676	17,561
attributes	19	122

To ensure a comprehensive assessment of the FedSarah algorithm's performance, this study employs multiple evaluation metrics across various datasets. Specifically, for the Synthetic and Fashion_MNIST datasets, it prioritizes accuracy (ACC) as the primary metric to gauge the model's classification performance. And additionally, it uses the HR@3 (Hit Rate at 3) and NDCG@3 (Normalized Discounted Cumulative Gain at 3) metrics for the Fashion_MNIST dataset, taking into account the small number of classes. The metrics HR@10 (Hit Rate at 10) and NDCG@10 (Normalized Discounted Cumulative Gain at 10) are used in processing the MovieLens-100K and ASSIST datasets. HR@10 serves as a measure of the recommendation system's ability to include the consumer's actual interactive behaviors (such as clicks or purchases) within the first 10 recommendations. It also provides insights into the system's effectiveness in recommending items of genuine interest to consumers. Conversely, NDCG@10 offers a more comprehensive evaluation metric. It not only takes into account whether the recommended items engage with consumers but also assesses the ranking of these interacting items within the recommendation list. This comprehensive approach enables a thorough evaluation of the recommendation system's performance. By adopting this holistic approach, encompassing diverse evaluation metrics across distinct datasets, we can accurately gauge the FedSarah algorithm's effectiveness in various scenarios, thereby providing a comprehensive assessment of its performance.

For a fair comparison of the effectiveness of the recommendations, the parameters of the three algorithms are controlled evenly on a single platform for the experiments. The platform setup is NVIDIA's v100-16G RAM 2-core CPU, CUDA version 11.0, and Tensorflow version tensorflow-gpu1.14. The unified settings for the experimental parameters are the same number of data batches, the same number of consumer clients picked for each round, the same maximum number of rounds ran for each consumer client, and the same number of servers. The number of global iterations is also the same, as is the learning rate.

B. Experimental performance comparison and analysis

First, the recommendation effectiveness of the FedAvg, FedProx, and FedSarah algorithms is compared on the Fashion_MNIST and two synthetic datasets, respectively. Here, the maximum number of consumer client running rounds is set to 10, the batch data size is set to 10, the learning rate is set to 0.01, and 10 randomly picked consumer clients from all the consumer clients are trained in each round. In Figure 3, the experimental results are displayed. FedAvg only tested on the Synthetic_iid dataset with smoother accuracy than the other two algorithms. The test accuracy varied widely and failed to converge in the other two datasets. The FedAvg algorithm may be stated to have significant space for improvement when the data are non-independent and homogeneously distributed. In other words, the data heterogeneity of consumer-centric personalized recommendation systems makes it impossible to produce personalized recommendations. In contrast, whether evaluated against either non-independent and identically distributed datasets or independently distributed datasets, the FedSarah and FedProx algorithms are reasonably stable. And as observed in Figure 3, both the rising curve of test accuracy and the falling curve of local training loss of FedSarah are less volatile than those of FedProx. This demonstrates that, even in complex scenarios, FedSarah is able to effectively balance the feature preferences of individual consumers with correction terms to produce better and more accurate product recommendations to consumers.

Tables III and IV show the accuracies of FedAvg, FedProx, and FedSarah algorithms for the first fifty rounds of testing on the Fashion_MNIST and Synthetic_1 datasets, respectively. According to the results, their performances are similar to those presented in Figure 3. FedSarah beats the FedProx algorithm in most circumstances on non-independently and identically distributed datasets, and it increases recommendation

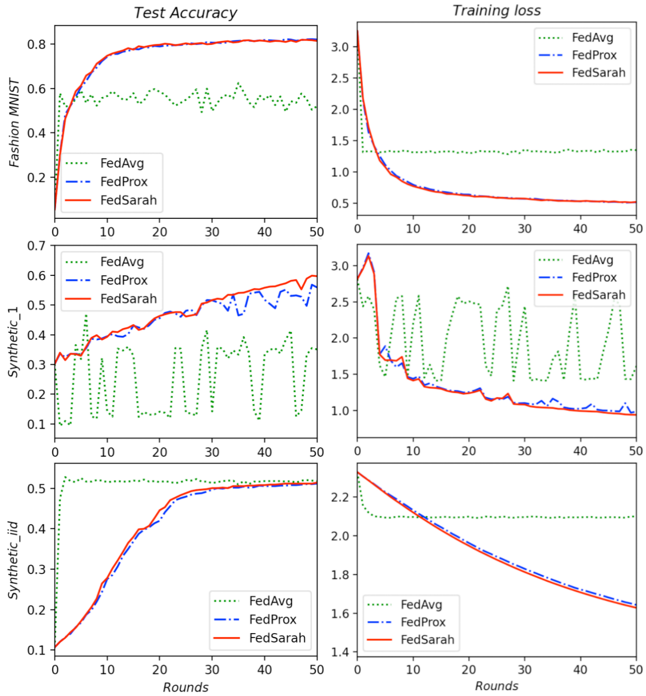

Fig. 3. Comparison of FedSarah with FedAvg and FedProx convergence results

accuracy by up to 5% over that of the FedProx algorithm. In Table V, the data are independently and identically distributed, and it appears that FedAvg can achieve 51% accuracy by the tenth round, while FedProx and FedSarah only achieve nearly 50% accuracy by the thirtieth round. In terms of convergence rate, FedAvg works better on independent and identically distributed datasets, although the accuracy differences between these three algorithms become smaller as the number of training iterations grows. In other words, the gaps in accuracy between these three algorithms become less and less as the number of iteration rounds keeps increasing.

TABLE III
THE ACCURACIES OF FEDAVG, FEDPROX, AND FEDSARAH ON FASHION_MNIST DATASET

Fashion_MNIST	FedAvg	FedProx	FedSarah
Round 10	0.55668	0.72297	**0.7288**
Round 20	0.59714	0.79154	**0.79508**
Round 30	0.59897	**0.80811**	0.79885
Round 40	0.56845	**0.816**	0.81588
Round 50	0.50422	**0.82171**	0.81725

TABLE IV
THE ACCURACIES OF FEDAVG, FEDPROX, AND FEDSARAH ON THE SYNTHETIC_1 DATASET

Synthetic_1	FedAvg	FedProx	FedSarah
Round 10	0.32195	**0.38745**	0.38376
Round 20	0.13191	0.45018	**0.45479**
Round 30	0.41605	**0.5166**	0.5083
Round 40	0.1107	0.54428	**0.55258**
Round 50	0.35516	0.56826	**0.5987**

TABLE V
THE ACCURACIES OF FEDAVG, FEDPROX, AND FEDSARAH ALGORITHMS ON THE SYNTHETIC_IID DATASET

Synthetic_iid	FedAvg	FedProx	FedSarah
Round 10	**0.51925**	0.24347	0.26459
Round 20	**0.51801**	0.41366	0.42608
Round 30	**0.51677**	0.49192	0.49813
Round 40	**0.51577**	0.50683	0.50807
Round 50	**0.51925**	0.51055	0.5118

To explore the effectiveness of these three algorithms for recommendation on the Synthetic_iid dataset, additional experiments are conducted. The batch data size is still set to 10, and 10 clients are randomly selected from all the consumer clients for training in each round. The maximum number of local training rounds is increased to 20, and the learning rate is set to 0.003 for a total of 400 rounds. In view of this, the number of rounds is a little larger in this experiment. For better comparison of the recommendation effects, the Synthetic_1 dataset is included, and a reduced learning rate is used to train the model with the goal of obtaining better personalized recommendations. The results are shown in Figure 4. After roughly 100 rounds on the Synthetic_iid dataset, the accuracies of the three algorithms are almost the same. And after that, FedProx's and FedSarah's accuracies begin to steadily surpass those of FedAvg. Combined with the performances on the Synthetic_1 dataset, FedSarah has less volatility during convergence than that of FedProx, and its reduction of training loss is also better than that of FedProx. So it can be claimed that in the heterogeneous environment of a consumer-centric personalized recommendation system, FedSarah not only increases the convergence accuracy but also makes the convergence process more stable compared to that of FedProx. As a result, it is clear that FedSarah can make personalized recommendations to consumers more reliably and precisely.

Considering the varying computing power of different consumer clients, we analyze the influence of uneven consumer computing power on the accuracy of personalized recommendations in a consumer-centric personalized recommendation system. Utilizing the hardware and software parameters of consumer client devices to assess their computational capacity is a typical method of gauging mathematical capability. In this study, the number of rounds of local updates on the consumer side is controlled using different γ values to mimic consumers with varying computing power. Firstly, the γ parameter is modified on the Fashion_MNIST, Synthetic_1, and Synthetic_iid datasets to replicate the condition of consumer clients with varied computing power. $l0, l1$, and $l2$ represent consumer clients with varying computing powers, and their γ values become larger and larger as they increase. This means the computing power of consumer clients is decreasing more and more. The experimental batch data size is 10, and 10 consumer clients are selected in each training round. Each consumer client may attend a maximum of 20 training sessions, and the learning rate is fixed at 0.003. The experimental results are shown in Figure 5. A comparison of the three γ values' performance on the three datasets indicates that the red line

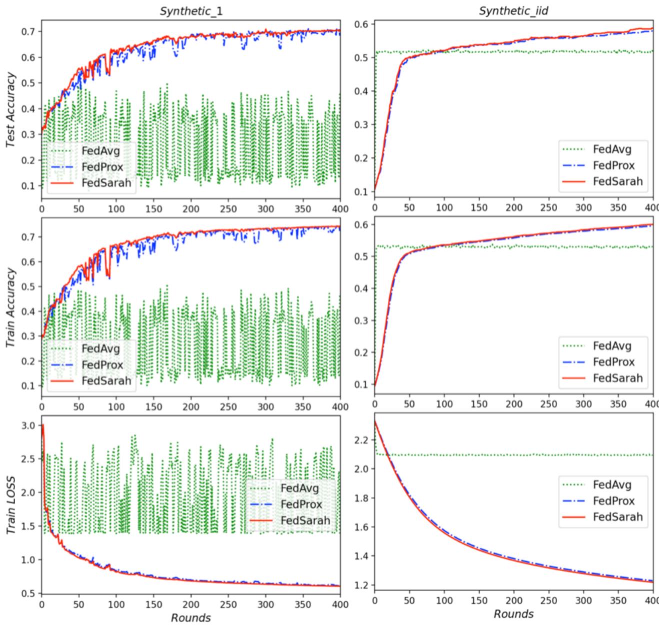

Fig. 4. Comparison of FedSarah, FedAvg and FedProx stability graphs

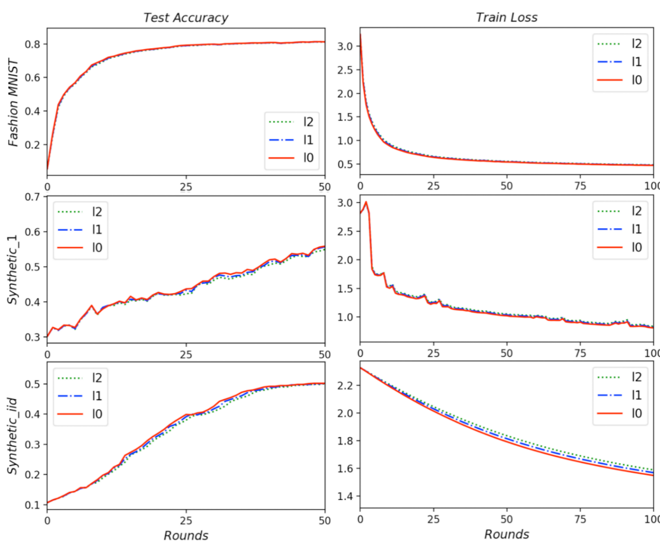

Fig. 5. Impact of γ parameter on the global model efficiency

representing $l0$ is on average higher in test accuracy than that of the blue line representing $l1$. And that of the blue line representing $l1$ is likewise, on average, higher in test accuracy than that of the green line representing $l2$. It proves that FedSarah converges faster and also has greater test accuracy while decreasing local losses faster when the γ value is lower. It is easy to find that, after 50 rounds, the γ value has very little influence on its convergence accuracy, with a maximum value of no more than 0.9%. This supports the proposed design idea that, in the complex scenario of a consumer-centric personalized recommendation system, for consumer clients with insufficient computing power, the number of rounds of local computation can be reduced by setting a larger γ to reduce the pressure on their local computation. So that the consumer clients will not be dropped because they

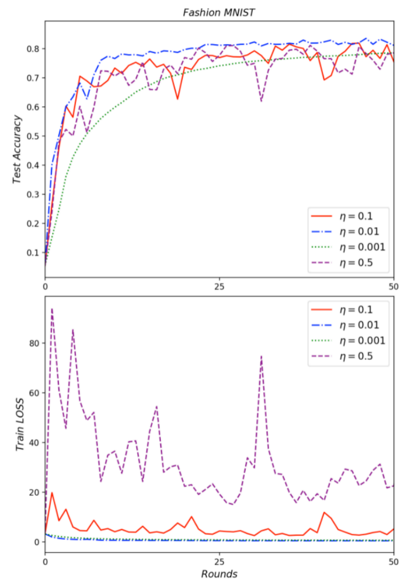

Fig. 6. Impact of learning rate on the global model performance

are unable to execute the local update at the specified time. This reduces the impact of consumer clients with different computing powers on the global model aggregation. Of course, this method can also be used to improve the model's tolerance for consumer clients with uneven computing power through the prediction of computing power in order to allow more consumer clients to participate in the model's updating. So that it can improve the model's scalability while ensuring the model's personalized recommendation has high accuracy and low latency.

Afterwards, the influence of learning rate on global model recommendation performance is further tested on the Fashion_MNIST dataset. Figure 6 depicts the experimental results. In the experiments, the learning rate is set to 0.5, 0.1, 0.01, and 0.001, respectively, and the batch data size is set to 10, 10 consumer clients are selected in each round, the γ value is $l0$, and the maximum number of local trainers is 20, for a total of 50 rounds. It can be seen from Figure 6 that the test accuracy performance of the blue line with a learning rate of 0.01 is the best, while that of the red line with a learning rate of 0.1 and that of the purple line with a learning rate of 0.5 are comparable. However, the red line has better performance and decreases faster in terms of training loss reduction. The green line with a learning rate of 0.001 has a slower and less precise convergence than that of the rest of the lines, but performs well in lowering the training loss. We can infer from these results that, among the four learning rate values, FedSarah has the highest accuracy when the learning rate is 0.01, and its training loss is reduced faster. When the learning rate increases, the training loss reduction fluctuates too much and is very unstable. The training loss is effectively minimized when the learning rate is low enough. However, there is a drop in performance in terms of accuracy. It may be considered that the learning rate is not as low as is feasible for FedSarah, but the ideal value is between 0.1 and 0.001. At this point, we can make the most helpful recommendations.

TABLE VI
FEDAVG, FEDPROX AND FEDSARAH PERFORMANCE RESULTS AT FASHION_MNIST

	FedAvg	FedProx	FedSarah
ACC	0.598	**0.821**	0.817
HR@3	0.907	0.941	**0.962**
NDCG@3	0.851	0.897	**0.902**

TABLE VII
FEDAVG, FEDPROX AND FEDSARAH PERFORMANCE RESULTS AT MOVIELENS-100K

	FedAvg	FedProx	FedSarah
ACC	0.398	0.406	**0.429**
HR@10	0.753	0.766	**0.812**
NDCG@10	0.505	0.523	**0.638**

TABLE VIII
FEDAVG, FEDPROX AND FEDSARAH PERFORMANCE RESULTS AT ASSIST

	FedAvg	FedProx	FedSarah
ACC	0.691	0.701	**0.726**
HR@10	0.891	0.902	**0.913**
NDCG@10	0.831	0.836	**0.855**

In Tables VI, VII, and VIII, we present the experimental results for three algorithms: FedAvg, FedProx, and FedSarah, applied to the Fashion_MNIST, Movielens-100K, and ASSIST datasets. On the Fashion_MNIST dataset, FedSarah demonstrates excellent performance. In terms of accuracy, FedSarah achieved an impressive score of 0.817 compared to FedAvg (0.598) and FedProx (0.821), highlighting its significant strength in accurately capturing user interest in fashion items and predicting their interactions. For the HR@3 criterion, FedSarah achieves a score of 0.962, significantly outperforming FedAvg (0.907) and FedProx (0.941), highlighting FedSarah's ability to more accurately reflect actual user preferences when generating recommendation lists. In addition, based on the NDCG@3 criterion, FedSarah obtains a high score of 0.902, which is significantly better than FedAvg (0.851) and FedProx (0.897). This indicated that FedSarah not only performs excellently in terms of the number of clicks on the recommended items but also exhibits a more outstanding performance when the ranking of the items is taken into account. So that it offers a more comprehensive assessment of the personalized recommendation of fashion products and provides a more comprehensive evaluation. On the Movielens-100K dataset, FedSarah demonstrates a notable superiority in terms of accuracy, achieving an impressive score of 0.429 compared to FedAvg (0.398) and FedProx (0.406). This highlights FedSarah's capability to significantly enhance recommendation accuracy by more precisely capturing consumer interests and predicting their interactions with films. Considering the HR@10 criterion, FedSarah excels with a score of 0.812, surpassing FedAvg (0.753) and FedProx (0.766). This underscores that the recommendation lists generated by FedSarah are more likely to contain items that consumers genuinely interact with, reflecting its superior performance in terms of hit rate. Furthermore, judging by the NDCG@10 criterion, FedSarah achieves a substantial score of 0.638, significantly outperforming FedAvg (0.505) and FedProx (0.523). This signifies that FedSarah exhibits superior performance not only in item hits but also in considering the ranking of recommended items, providing a more comprehensive evaluation. On the ASSIST dataset, FedSarah maintains its high standard by achieving a higher accuracy of 0.726, compared to 0.691 for FedAvg and 0.701 for FedProx. This indicates that FedSarah excels at accurately capturing intrinsic patterns and trends, making it adept at predicting learner behavior. In terms of the HR@10 criterion, FedSarah leads with a score of 0.913, surpassing FedAvg (0.891) and FedProx (0.902). This implies that FedSarah's recommendation lists are more likely to include items that learners genuinely interact with within the top 10 recommendations. Even when considering the NDCG@10 criterion, FedSarah maintains its edge with a score of 0.855, slightly higher than FedAvg (0.831) and FedProx (0.836). This shows that FedSarah not only excels in item hits but also maintains a strong performance in item rankings. In summary, based on the experimental results, FedSarah exhibits superior accuracy, hit rate, and item ranking capabilities, making it exceptionally suitable for personalized recommendation scenarios.

TABLE IX
THE TIME COMPLEXITY OF THE THREE ALGORITHMS

Algorithm	Time Complexity
FedAvg	O(T*K*(L+d))
FedProx	O(T*K*(L+d+l))
FedSarah	O(T*K*(L+d+l))

Finally, the time complexity of the three algorithms is shown in Table IX. The time complexity of federated learning algorithms typically involves a number of factors, such as communication overhead, local computation, aggregation computation, and global model updating. In Table IX, we denote the local computation as O(L), where L represents the number of local training rounds, the communication overhead as O(K), where K represents the number of devices, the aggregation computation as O(d), where d represents the dimensionality of the model parameters, and the number of iteration rounds as O(T), where T represents the total number of training rounds. Since FedProx and FedSarah introduce additional computation, this overhead is denoted as O(l). The results reveal that both FedSarah and FedProx introduce nearly identical additional computational overhead, and FedSarah even aligns with the time complexity of FedAvg when the additional computation is zero. When considered alongside the previous experimental findings, it becomes evident that the FedSarah algorithm can significantly enhance experimental results without imposing a substantial computational burden. Consequently, it can be deduced that this algorithm is highly suitable for applications in the consumer electronics sector, where the need for rapid iterations and high accuracy is paramount.

V. Conclusion

This paper proposes FedSarah, a federated learning algorithm employing a recursive framework to enhance stochastic gradient estimation. It addresses the challenge of excessive communication latency in global model aggregation caused by data heterogeneity and computing power disparities in consumer-centric personalized recommendation systems. FedSarah optimizes communication efficiency by incorporating a correction term into the local computation at each consumer client, individualizing it for improved global model accommodation of participating consumers. The algorithm demonstrates theoretical linear convergence for strong convexity and sublinear convergence for general convexity. Applied to consumer electronics product promotion in heterogeneous consumer-centric personalized recommendation systems, FedSarah reduces communication frequency and latency while enhancing recommendation accuracy. The experimental results confirm FedSarah's ability to consistently and precisely adapt to consumer interests, ensuring data security and privacy, and ultimately boosting customer satisfaction through reduced communication latency.

References

[1] D. Javeed, M. S. Saeed, I. Ahmad, P. Kumar, A. Jolfaei, and M. Tahir, "An intelligent intrusion detection system for smart consumer electronics network," *IEEE Transactions on Consumer Electronics*, 2023.

[2] G. Li and Q. Chen, "Exploiting explicit and implicit feedback for personalized ranking," *Mathematical Problems in Engineering*, vol. 2016, pp. 1–11, 2016.

[3] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl, "Grouplens: Applying collaborative filtering to usenet news," *Communications of the ACM*, vol. 40, no. 3, pp. 77–87, 1997.

[4] C. Jebarajakirthy and A. Shankar, "Impact of online convenience on mobile banking adoption intention: A moderated mediation approach," *Journal of Retailing and Consumer Services*, vol. 58, p. 102323, 2021.

[5] P. K. Donta, S. N. Srirama, T. Amgoth, and C. S. R. Annavarapu, "Survey on recent advances in iot application layer protocols and machine learning scope for research directions," *Digital Communications and Networks*, vol. 8, no. 5, pp. 727–744, 2022.

[6] C. K. Dehury, P. K. Donta, S. Dustdar, and S. N. Srirama, "Ccei-iot: Clustered and cohesive edge intelligence in internet of things," in *2022 IEEE International Conference on Edge Computing and Communications (EDGE)*. IEEE, 2022, pp. 33–40.

[7] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[8] Y. Kim, J. Sun, H. Yu, and X. Jiang, "Federated tensor factorization for computational phenotyping," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 887–895.

[9] J. Lee, J. Sun, F. Wang, S. Wang, C.-H. Jun, X. Jiang *et al.*, "Privacy-preserving patient similarity learning in a federated environment: development and analysis," *JMIR medical informatics*, vol. 6, no. 2, p. e7744, 2018.

[10] K. Sozinov, V. Vlassov, and S. Girdzijauskas, "Human activity recognition using federated learning," in *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*. IEEE, 2018, pp. 1103–1111.

[11] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018.

[12] Y. M. Saputra, D. T. Hoang, D. N. Nguyen, E. Dutkiewicz, M. D. Mueck, and S. Srikanteswara, "Energy demand prediction with federated learning for electric vehicle networks," in *2019 IEEE global communications conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[13] W. Yang, Y. Zhang, K. Ye, L. Li, and C.-Z. Xu, "Ffd: A federated learning based method for credit card fraud detection," in *Big Data–BigData 2019: 8th International Congress, Held as Part of the Services Conference Federation, SCF 2019, San Diego, CA, USA, June 25–30, 2019, Proceedings 8*. Springer, 2019, pp. 18–32.

[14] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic federated learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 4615–4625.

[15] B. Liu, L. Wang, M. Liu, and C.-Z. Xu, "Federated imitation learning: A novel framework for cloud robotic systems with heterogeneous sensor data," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3509–3516, 2020.

[16] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[17] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and robust federated learning through personalization," in *International Conference on Machine Learning*. PMLR, 2021, pp. 6357–6368.

[18] J. Oh, S. Kim, and S.-Y. Yun, "Fedbabu: Towards enhanced representation for federated image classification," *arXiv preprint arXiv:2106.06042*, 2021.

[19] Z. Qu, Y. Tang, G. Muhammad, and P. Tiwari, "Privacy protection in intelligent vehicle networking: A novel federated learning algorithm based on information fusion," *Information Fusion*, vol. 98, p. 101824, 2023.

[20] X. Fu, B. Zhang, Y. Dong, C. Chen, and J. Li, "Federated graph machine learning: A survey of concepts, techniques, and applications," *ACM SIGKDD Explorations Newsletter*, vol. 24, no. 2, pp. 32–47, 2022.

[21] Y. Li, X. Wang, R. Zeng, P. K. Donta, I. Murturi, M. Huang, and S. Dustdar, "Federated domain generalization: A survey," *arXiv preprint arXiv:2306.01334*, 2023.

[22] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "Deepfm: a factorization-machine based neural network for ctr prediction," *arXiv preprint arXiv:1703.04247*, 2017.

[23] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 689–698.

[24] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," in *The world wide web conference*, 2019, pp. 417–426.

[25] G. Beigi, A. Mosallanezhad, R. Guo, H. Alvari, A. Nou, and H. Liu, "Privacy-aware recommendation with private-attribute protection using adversarial learning," in *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020, pp. 34–42.

[26] K. Muhammad, Q. Wang, D. O'Reilly-Morgan, E. Tragos, B. Smyth, N. Hurley, J. Geraci, and A. Lawlor, "Fedfast: Going beyond average for faster training of federated recommender systems," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1234–1242.

[27] G. Lin, F. Liang, W. Pan, and Z. Ming, "Fedrec: Federated recommendation with explicit feedback," *IEEE Intelligent Systems*, vol. 36, no. 5, pp. 21–30, 2020.

[28] F. K. Khan, A. Flanagan, K. E. Tan, Z. Alamgir, and M. Ammad-Ud-Din, "A payload optimization method for federated recommender systems," in *Proceedings of the 15th ACM Conference on Recommender Systems*, 2021, pp. 432–442.

[29] C. Wu, F. Wu, Y. Cao, Y. Huang, and X. Xie, "Fedgnn: Federated graph neural network for privacy-preserving recommendation," *arXiv preprint arXiv:2102.04925*, 2021.

[30] O. Shamir, N. Srebro, and T. Zhang, "Communication-efficient distributed optimization using an approximate newton-type method," in *International conference on machine learning*. PMLR, 2014, pp. 1000–1008.

[31] S. J. Reddi, J. Konečný, P. Richtárik, B. Póczós, and A. Smola, "Aide: Fast and communication efficient distributed optimization," *arXiv preprint arXiv:1608.06879*, 2016.

[32] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[33] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th international conference on machine learning (icml-03)*, 2003, pp. 928–936.

[34] [Online]. Available: https://grouplens.org/datasets/movielens/100k

[35] [Online]. Available: https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010data/non_skill-builder-data-2009-2010