TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214
DOI: 10.26599/TST.2025.9010175

Holistic Evaluation Metrics: Use Case Sensitive Evaluation Metrics for Federated Learning

Yanli Li[1], Jehad Ibrahim[1], Huaming Chen*, Dong Yuan, and Kim-Kwang Raymond Choo*

Abstract: A large number of federated learning (FL) algorithms have been proposed for different applications and from varying perspectives. However, the evaluation of such approaches often relies on a single metric (e.g., accuracy). Such a practice fails to account for the unique demands and diverse requirements of different use cases. Thus, how to comprehensively evaluate an FL algorithm and determine the most suitable candidate for a designated use case remains an open question. To mitigate this research gap, we introduce the Holistic Evaluation Metrics (HEM) for FL in this work. Specifically, we collectively focus on three primary use cases, which are Internet of Things (IoT), smart devices, and institutions. The evaluation metric encompasses various aspects including accuracy, convergence, computational efficiency, fairness, and personalization. We then assign a respective importance vector for each use case, reflecting their distinct performance requirements and priorities. The HEM index is finally generated by integrating these metric components with their respective importance vectors. Through evaluating different FL algorithms in these three prevalent use cases, our experimental results demonstrate that HEM can effectively assess and identify the FL algorithms best suited to particular scenarios. We anticipate this work sheds light on the evaluation process for pragmatic FL algorithms in real-world applications.

Key words: Federated Learning (FL), Decentralized Machine Learning (DML), Evaluation Metrics, Collaboration Application

1 Introduction

Federated learning (FL) is a trending distributed learning paradigm that enables many clients to train a machine learning model collaboratively while keeping the training data decentralized [1, 2]. The underpinning privacy protection nature partly resulted in its application in various real-world scenarios, including *smartphones* (smart devices [3]), *institutional* settings [4], and *Internet of Things* (IoT) environment [5]. For instance, Google implements the intelligent keyboard "Gboard" on its Pixel smartphones, and Apple has developed its virtual assistant "Siri" for smart i-devices. Studies such as [5] have explored the deployment of FL in health and medical institutions for achieving intelligent diagnosis and treatment. Furthermore, the research in [6] demonstrates FL's significant potential in IoT scenarios, including vehicular traffic planning, unmanned aerial vehicles, and the smart city.

Different FL algorithms have been developed for various real-world applications [7] with different

• Yanli Li is with School of Artificial Intelligence and Computer Science, Nantong University, and also with the School of Electrical and Computer Engineering, The University of Sydney (E-mail: yanli.li@sydney.edu.au)
• Jehad Ibrahim, Huaming Chen and Dong Yuan are with the School of Electrical and Computer Engineering, The University of Sydney (E-mail:jibr7656@uni.sydney.edu.au, {huaming.chen, dong.yuan}@sydney.edu.au).
• Kim-Kwang Raymond Choo is with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX, TX 78249, USA. (E-mail: raymond.choo@fulbrightmail.org).
∗ To whom correspondence should be addressed.
Manuscript received: 2025-07-23; revised: 2025-09-14; accepted: 2025-11-10

requirements, ranging from reduced communication costs (e.g., FedAvg [8]) to non-independent identical distribution (non-IID) settings (e.g., FedDyn [9], SCAFFOLD [10], CFLGT [11]) and personalized local model (e.g., MAML, ProtoNet [12]), etc. Given the diversity of FL algorithms, it is no surprise that selecting an appropriate candidate for a specific use case is an important research trend [8]. To inform the selection, existing studies typically assess and compare FL algorithms based on a single metric, such as accuracy or loss value. While such evaluation indexes can partially reflect the performance of FLff algorithms from particular perspectives, they fall short in providing a comprehensive assessment of the overall model performance [8].

Furthermore, evaluations based on a single metric fail to reflect the practical diversity of FL deployments, where different application scenarios often impose distinct requirements and optimization priorities. Such oversimplified assessments may therefore lead to biased or unreliable evaluation outcomes. To address this research gap, we propose a novel Holistic Evaluation Metrics (HEM) framework and a scenario-adaptive evaluation pipeline that dynamically aligns the importance of different performance metrics with the specific requirements of diverse FL application contexts. This adaptive design enables a more comprehensive and context-sensitive evaluation of FL algorithms, setting HEM apart from existing static evaluation approaches.

Given the unique performance requirements of each FL use case, we start by highlighting the three most representative contexts, namely smartphones (smart devices), institutions, and IoT [13, 14]. Our proposed evaluation framework, the HEM index, goes beyond a singular metric by encompassing multiple key components, including Client Accuracy, Convergence, Computational Efficiency, Fairness, and Personalization (for personalized FL-PFL algorithms), ensuring a comprehensive assessment. The HEM for any FL algorithm is generated by a weighted average of these component performance indices, where the weights assigned are proportional to the particular needs identified (referred to as importance vectors) for the target application. Specifically, we identify that accuracy and fairness are the primary performance requirements for FL applications in IoT and institutional contexts [8, 15]. In contrast, convergence and computational efficiency are the main focus in smartphone-related applications.

In this study, we evaluate a range of FL algorithms and personalization methods across these identified cases using our proposed HEM and the corresponding evaluation pipeline. The experimental results demonstrate that the HEM can effectively evaluate and differentiate FL algorithms, aiding in the algorithm selection for target use cases. While we currently focus on three FL use cases in this paper, our HEM could be easily extended to other application scenarios through further investigation into the importance of different metric components.

The main contributions of our work are:

- We establish a comprehensive set of metrics, including accuracy, convergence, computational efficiency, fairness, and personalization for organizing the evaluation metrics.
- We identify the specific needs of different use cases and assess the importance of each corresponding component accordingly.
- We propose the HEM index based on the evaluation components and importance identified, achieving comprehensive and effective evaluation for an FL algorithm in the designated use cases.

This work has substantially extended our previous study [16]. Specifically, we extend Section 2.1 for a more comprehensive survey on the FL application, Section 3.4 about the detailed HEM evaluation process, and Section 5 with the insight of the performance trade-off by introducing personalization.

The rest of this paper is organized as follows. The next section provides a related works survey of existing federated learning applications, algorithms, and evaluation metrics. Section 3 describes the proposed HEM, followed by the experimental results and related discussions in Section 4 and 5. The last section concludes the paper.

2 Related Works

2.1 Federated Learning Applications

To utilize data isolated at client locations without compromising privacy, numerous applications have been deployed employing FL architecture. These applications typically fall into three categories, depending on the use cases. They include applications for smart phones (smart devices) [17, 18], institutions [19, 20], and IoT [21–23].

Firstly, Google's successful deployment of its FL-based user input prediction keyboard (Gboard) has sparked significant interest in developing FL applications for smart devices. This application category of smart devices now includes not only text prediction of user input but also emoji prediction, monitoring health conditions (such as depression detection) [17], and offering energy efficiency plan recommendations to users [18]. Secondly, FL's capabilities in preserving privacy have led to its adoption in data-sensitive institutional settings. A recent study [19] introduces a para-medicine application designed to predict the histological response to neoadjuvant chemotherapy (NACT) in early-stage breast cancer patients. FL has facilitated the utilization of clinical information while ensuring data is securely maintained behind hospital firewalls. To support the treatment for patients with COVID-19, another study [24] proposes an FL-based prediction model, named EXAM, which successfully uses the data from 20 institutes across the globe without compromising privacy. Thirdly, FL is being applied in the IoT sector to address the limitations of high storage costs, together with privacy concerns associated with the traditional ecosystem of centralized over-the-cloud learning and IoT platforms. To offer diversified electricity services, the study [25] introduces an FL application that identifies electricity consumer characteristics. Specifically, privacy-preserving principal component analysis (PCA) is utilized to extract features from smart meter data, while an artificial neural network is trained in a federated setting with various weighted averaging strategies.

2.2 Federated Learning Algorithms

FL allows multiple participants to collaboratively train a shared model without sharing their raw data. Typically, a FL training round includes three steps: the server first broadcasts the current global model to all participants. Subsequently, each participant locally updates the global model using their own private data, sending the updated model back to the server once the established training criteria are satisfied. Finally, the server aggregates the models received using a designated FL algorithm and finalizes the current training round [26, 27].

As a key FL algorithm, FedAvg [28] efficiently aggregates the client gradients while minimising the communication costs. In the FedAvg approach, the server computes a weighted average of the clients' models, where the weights are proportional to the size of the training data each client possesses. Despite reducing the communication bottleneck in FL, FedAvg stuggles with handling non-IID data and heterogeneous data, limiting its deployment in real-world scenarios. To address these shortcomings and further improve FL performance, FedDyn [9] and SCAFFOLD [10] have been developed. The algorithm FedDyn builds on the concept of adding a penalty term to the objective function introduced by FedProx, but dynamically changes the value of the penalty term by using a novel dynamic regularization method to ensure that the minima of the global and local objective functions are consistent [9]. SCAFFOLD is an algorithm that builds on FedAvg by attempting to improve its convergence limitation when trained on non-IID data. It corrects the "client drift" issue by leveraging variant reduction.

In addition to ensuring identical model functions for all participants, there is a growing research focus on developing personalized FL algorithms and providing customized learning performances. The study [29] proposes two algorithms, PFLDyn and PFLScaf, which apply the gradient correction methods (FedDyn and SCAFFOLD) to correct bias in a meta-model and tailor personalized client models. Two meta-learning approaches are used to implement the algorithms: MAML and ProtoNet [29]. The algorithms present convergence guarantees for convex and nonconvex meta objectives. Further, the research [12, 30, 31] propose personalized versions of the FedAvg algorithm. In general, personalized federated learning aims to balance global collaboration and local adaptation, enabling each client to obtain a model that better fits its own data distribution while still benefiting from shared knowledge [31]. This paradigm acknowledges the inherent heterogeneity among clients and seeks to improve both overall performance and individual satisfaction by jointly leveraging shared representations and client-specific customization.

2.3 Federated Learning Evaluation Metrics

Currently, the evaluation metrics used in FL generally derive from those established in traditional centralized machine learning (CML) concepts. The development of specific metrics tailored for FL is still in its nascent stages. Commonly, metrics like accuracy or loss value, which measure the percentage of correct prediction on the testing data or quantify

FL/PFL Algorithms	Evaluation Criteria			
	Accuracy	**Convergence**	**Complexity**	**Fairness**
FedAvg	✓	✓	×	×
FedAvg (MAML)	×	✓	×	×
FedAvg (Proto)	✓	✓	×	×
FedDyn	✓	✓	✓	×
FedDyn (MAML)	✓	✓	×	×
FedDyn (Proto)	✓	✓	×	×
SCAFFOLD	✓	✓	✓	×
SCAFFOLD (MAML)	✓	✓	×	×
SCAFFOLD (Proto)	✓	✓	×	×

Table 1 Criteria used for different FL algorithms evaluations.

the difference between predictions and actual results, are used to evaluate both FL and CML algorithms [29, 32]. In addition, since FL training relies on regular communication between the server and clients, some research evaluates algorithm performance based on communication efficiency [33], considering the factors such as the number of communication rounds, parameters, and message sizes required to train an FL model [29].

From a system performance perspective, FL algorithms can be evaluated based on their network and hardware efficiency. Typical metrics include CPU/GPU utilization, energy consumption (joules per round), and device idle time. Higher GPU utilization indicates more efficient parallel computation [34], whereas prolonged idle time often reveals synchronization overhead [35]. Lower energy consumption [36] further signifies improved sustainability and cost efficiency, offering a comprehensive view of the algorithm's computational performance. As FL involves multiple typical participants, several studies [37] have made efforts to reduce the performance gap between different clients or groups. To measure such a gap, Equalized Odds Difference (EOD) and Statistical Parity Difference (SPD) are commonly employed [38]. Specifically, EOD assesses whether different groups have similar true and false positive rates, capturing fairness conditional on the ground truth, while SPD measures the difference in overall positive prediction rates between groups, reflecting unconditional outcome fairness.

We broadly review the key FL algorithms (i.e., FedAvg, FedDyn, and SCAFFOLD) and delve into two meta-learning techniques (i.e., MAML and Proto) that can adapt these FL algorithms into personalized FL (PFL) algorithms. While these algorithms could effectively address specific weaknesses of the original algorithm, the researchers used to evaluate and demonstrate improvements using a single metric. This approach often makes the trade-offs involved with each algorithm unclear.

Thus, in this work, we propose the Holistic Evaluation Metric (HEM), aiming to provide an effective and comprehensive evaluation of FL algorithms and aid in selecting the most appropriate FL algorithm for diverse real-world use cases. Table 1 illustrates the criteria used for different FL algorithm evaluations. We note that a recent study, FedEval [39], has been proposed to provide a multidimensional evaluation platform for FL algorithms. However, the motivation and design philosophy of FedEval and our proposed HEM framework are fundamentally different. Specifically, FedEval focuses on offering a comprehensive survey and encapsulation of existing evaluation metrics within a unified benchmarking platform. In contrast, the HEM framework is motivated by the need to dynamically aggregate evaluation metrics with scenario-specific importance weights, thereby enabling adaptive and context-aware evaluation across heterogeneous application environments.

3 Holistic Evaluation Metrics

In this section, we present Holistic Evaluation Metrics (HEM) to mitigate the research gap carried by the current simplistic evaluation metric and provide an effective evaluation method for FL algorithms

in various real-world use cases. Recognizing that different use cases may prioritize different performance aspects, we start by identifying the most representative FL use cases. We then determine the principal evaluation perspectives, and finally organize the HEM by developing the formula for the HEM index.

3.1 Federated Learning Use Cases

FL has revolutionized traditional ML by facilitating training on decentralized data while addressing crucial privacy concerns in privacy-sensitive applications where training data is distributed across multiple edge devices [40, 41]. Currently, a large number of service providers have adopted the FL method, which has found a significant role in different domains, ranging from smart devices to institutions and IoT. Herein, we explore the three representative real-world application use cases of FL, drawing on insights from the applications and characteristics outlined in recent studies [1, 42].

3.1.1 Smartphones (Smart Devices)

Smart devices, including smartphones and laptops, significantly benefit humans and have become integral to our daily lives. While providing information to users, these devices also collect user information and usage patterns. Applications such as next-word prediction, face detection, and voice recognition [43] could greatly benefit from these collected and recorded data from users. However, data gathering in a centralized location to support traditional centralized ML poses challenges due to user privacy concerns and legal restrictions. FL addresses these issues by facilitating the collaborative learning of ML models across a wide array of mobile devices. The smartphone participants can retain the data locally, only sharing the model update with the server via the cellular or wireless network. By leveraging FL, on the one hand, the vast amounts of user data ensure both high learning performance and satisfactory service provision; on the other hand, clients benefit from personalized functions through their participation. For instance, providers of next-word predictor applications can employ FL to train a model with the historical text data from users across a broad network of smart devices. In return, clients benefit from more accurate prediction results that are customized to their individual user habits [44].

3.1.2 Institutions

Existing institutions maintain large amounts of records and information on their clients to deliver services [5]. For example, healthcare institutions store diagnostic logs of patients to inform treatment strategies, while financial institutions keep records of clients' bank statements to make lending decisions. While such data could greatly improve training datasets and boost model performance, these institutions are frequently subject to strict privacy regulations and ethical considerations, necessitating that data remains localized [45]. This limitation hinders institutions' ability to utilize data from other organizations or even from their client data, ultimately constraining their potential to offer enhanced intelligent services. FL presents an innovative solution for these use cases by ensuring privacy and enabling institutions to participate in secure collaborations. Within an FL system, institutions act as "client" nodes sharing only the model updates during each learning iteration, thus maintaining data privacy within their boundaries [45]. Furthermore, vertical FL technology allows the deployment of an FL model in scenarios where participants share the same sample ID space but possess different feature sets, effectively accommodating collaboration between institutions even from different domains.

3.1.3 Internet of Things (IoT)

The IoT ecosystem consists of a network of interconnected devices, such as wearable devices [46], autonomous vehicles [47], and smart homes [48], all equipped with sensors to collect, process, and act on real-time data. For example, autonomous vehicles rely on continually updated models for safe navigation, taking into account traffic conditions, construction sites, and pedestrian movements. However, building accurate and up-to-date models can be challenging due to the privacy concerns associated with data and the limited connectivity of each device [49]. Moreover, the large volume of data generated by IoT devices every second further brings difficulty to the traditional centralized ML. To this end, FL algorithms are considered a viable alternative, which adapt IoT devices to dynamic environments, enabling efficient model training while preserving user privacy. When deploying FL in an IoT use case, each interconnected device can act as a client contributing partially or fully to the model training, depending on its computing resources [50]. A trustworthy cloud node serves as the server, orchestrating the entire training process.

3.1.4 Other Use Cases

Beyond the three representative use cases discussed above, real-world FL deployments may also encompass additional scenarios (Use Cases) such as edge computing environments and specific learning paradigms, including federated reinforcement learning and federated transfer learning. In these cases, the core evaluation metrics (such as accuracy, convergence, computational complexity, and fairness) remain applicable to assess model performance. However, additional components and their corresponding importance levels can be further defined and integrated into HEM based on domain experts' identification of scenario-specific requirements, such as communication efficiency for edge computing or reward stability for reinforcement learning. This flexibility ensures that HEM can be readily extended and adapted to emerging FL paradigms and deployment contexts.

3.2 Evaluation Metric Components

Rather than relying solely on a single performance metric, such as accuracy or loss value, we organize the HEM through multiple components to provide a more comprehensive evaluation. Specifically, these evaluation metric components encompass client accuracy, convergence, computation efficiency, fairness, and personalization for PFL.

Evaluation Metric Component 1: Client Accuracy
Accuracy, as the most intuitive indicator, is widely used in the evaluation of existing ML and FL algorithms to assess model learning performance [51]. Within the FL context, the global model used to be evaluated using testing data after each iteration's aggregation. The percentage of accurate predictions is then calculated as accuracy, serving as a measure to demonstrate the overall learning performance. However, due to the heterogeneity of participants' data and devices, their learning performance can vary significantly. Therefore, in the proposed HEM, we leverage client accuracy over global model accuracy, calculating it as the mean accuracy across clients per round. Client Accuracy reflects the utility and learning performance of the FL algorithm for individual clients under the given learning task and data distribution. Specifically, a higher Client Accuracy indicates greater satisfaction among clients regarding the predictions generated by their models. It essentially provides a clear measure of the FL algorithm's effectiveness and ability to meet the varied needs and expectations of participating clients.

Although this study primarily focuses on benign FL scenarios, the accuracy-based metric can also be extended to adversarial settings [41]. In such cases, the degradation rate of accuracy (i.e., the decline in client or global accuracy under attack conditions compared with benign training) can serve as an indicator of the algorithm's robustness and resilience. This extension provides a practical way to quantify performance stability when the FL system is exposed to malicious participants or data poisoning attacks.

Evaluation Metric Component 2: Convergence
The FL training heavily relies on the regular models (or gradients) exchange between clients and the server to perform the learning process. An algorithm that requires numerous learning rounds to achieve convergence may introduce high communication costs and exacerbate the bottleneck. To this end, we introduce the Convergence component in the HEM. The Convergence component reflects the time and computational resources an FL algorithm consumes by measuring the number of communication rounds an FL algorithm requires to achieve a predetermined target accuracy for clients. A higher convergence score indicates a heavier demand for communication, computational, and temporal resources, potentially taxing the devices involved. In our study, we set the target accuracy at 80%*, a benchmark deemed satisfactory for model performance across all investigated use cases. Additionally, we normalize the Convergence value on a scale from 0 to 1000, where 0 represents optimal performance and 1000 indicates the worst outcome. The normalized outcome will act as the Convergence index for the FL algorithm.

Evaluation Metric Component 3: Computation Efficiency
While HEM utilizes the Convergence component to represent the time cost by counting the communication rounds, the training time per round can vary across different FL algorithms, and finally drives an impact on the learning efficiency. To this end, we further include Computational Efficiency in HEM to introduce clock time cost evaluation. Specifically, the Computational Efficiency component reflects the amount of time or

*The accuracy threshold is determined based on general performance requirements but can be adjusted as needed for specific applications.

Use Cases	IoT	Smart devices (Smartphone)	Institution
Accuracy	High	Moderate	High
Convergence	Low	High	Low
Computational Efficiency	High	High	Low
Fairness	High	Moderate	High

Table 2 Holistic evaluation metric component importance for IoT, smartphone, and institution.

memory required for a given threshold but emphasizes the wall-clock time. In the HEM, the Computational Efficiency is measured using a metric called Time to Accuracy (TTA). TTA merges the concepts of Convergence and Simulation Time, providing a view of the duration necessary for an algorithm to meet the set target accuracy or threshold. Besides, this component evaluates the balance between computational resource demand and time investment and demonstrates an algorithm's efficiency in resource utilization.

Evaluation Metric Component 4: Fairness

The Fairness reflects the level of the learning performance (model accuracy) difference that exists in the FL system across all participants. In HEM, the Fairness component is assessed by calculating (inversely proportional to) the entropy (H) of the Client Accuracy list. Formally, for a given accuracy list $L = \{a_1, ..., a_i\}, |L| = I$, where a_i denotes the accuracy of the ith client. The Fairness (F) of L is:

$$1/F(L) \propto H(L) = -\sum_{i=0}^{I} a_i log_e a_i \tag{1}$$

A lower entropy signifies a more equitable distribution of accuracy, while a higher entropy points to disparities in performance among clients. Due to the data and device heterogeneity, clients usually achieve different learning performances in FL systems. Introducing Fairness in the evaluation metric avoids overemphasizing the overall global model and ignores the participant's disadvantage. It illustrates the balance of accuracy among the clients, indicating the algorithm's ability to address the diverse capabilities and characteristics of participating devices, so-called fairness.

While this work primarily focuses on fairness among individual participants, we acknowledge that fairness in federated learning (FL) can also be defined at the group level, for instance, across demographic groups such as males and females. In such cases, fairness can be measured using the Equal Opportunity Difference (EOD) metric:

Equal Opportunity Difference (EOD): For samples belonging to the positive class ($Y = 1$), the probability of being correctly predicted as positive ($\hat{Y} = 1$) should be similar across groups characterized by different sensitive attributes F. The difference between these probabilities across two groups can be formulated as:

$$\begin{aligned} \Delta_{EOD} = P(\hat{Y} = 1|Y = 1, F = f_1) \\ -P(\hat{Y} = 1|Y = 1, F = f_2) \end{aligned} \tag{2}$$

A smaller value of Δ_{EOD} indicates a higher level of group fairness, implying that the model provides comparable opportunities for correct predictions across different groups.

Beyond the individual- and group-level definitions discussed above, the concept of fairness in FL can also be extended to incorporate statistical measures that capture performance disparities from different perspectives. Such measures may include distributional metrics or variance-based indicators that quantify deviations in model performance across clients or demographic groups. These statistical formulations provide complementary insights into fairness and can be seamlessly integrated into the HEM framework as alternative or supplementary evaluation components, depending on the characteristics and objectives of the specific use case.

Evaluation Metric Component 5: Personalization

In FL systems, clients may have various exceptions for the global model due to their different interests and use patterns. To evaluate the effectiveness of customization achieved through personalization methods, we introduce the Personalization component into the HEM for PFL algorithms. This component is calculated by comparing the Client Accuracy list with and without introducing personalization methods. Within HEM, Personalization is represented as the median percentage improvement in accuracy

across all clients when using the PFL algorithm compared to the original FL algorithm. A higher personalization index indicates the method's capability to enhance performance for individual clients' specific data distribution, demonstrating strong potential for delivering personalized functions in real-world applications.

3.3 Holistic Evaluation Metrics (HEM)

Now, we construct the HEM using the identified components of the evaluation metric. This subsection presents the organization of HEM across the three most representative use cases—IoT, smartphone, and institution; however, HEM is versatile and can be applied to any other use case.

The HEM is generated through a linear combination of each evaluation metric component value and represented as an index ranging from 0 to 1, with 1 denoting the ideal algorithm for a specific use case. Specifically, each evaluation metric component is assigned an importance level (namely, Importance Vector) between 0 and 1, considering the specific requirements of the use case. Then, the holistic index is computed as a weighted average of the evaluation metric component indexes, taking into account the significance of each component within the given use case.

Formally, the HEM is generated by following the Equation:

$$HEM = \sum_{i=1}^{N} Index_i \cdot \overset{UseC}{ImporV_i} \tag{3}$$

where $Index_i$ and $ImporV_i$ indicate the value (Index) of each evaluation metric component and its associated Importance Vector, respectively. $UseC$ demonstrates the target use case, and N indicates the amount of evaluation metric component. Due to the unique characteristic of each use case, the $\overset{UseC}{ImporV_i}$ should be different. TABLE. 2 illustrates the HEM component importance for IoT, smartphone and institution use cases, and we will provide detailed information and discussion in the following subsections.

3.3.1 HEM for IoT Use Cases

Given the advancements in FL and IoT technologies, heavy machinery (including autonomous vehicles and cranes) can now operate autonomously, significantly reducing the human workload. However, given their considerable volume, weight, and strength, any inaccuracies in the decisions made by these machines could lead to life-threatening outcomes. Therefore, maintaining a high level of accuracy for every single participant in IoT scenarios is crucial, justifying the High importance of the evaluation components - Client Accuracy and Fairness. Similarly, the Computational Efficiency component is also deemed as High important due to the limited computational resources of most IoT devices. We note that although some of these devices are allowed to train FL models in the background without impacting their primary functions, their reduced CPU power and memory capacity restrict the effectiveness of such background processes, which emphasizes the priority of Computational Efficiency. On the contrary, due to the IoT devices typically remaining idle for an extended duration, which could make the execution of numerous FL communication rounds [52], the importance of the model Convergence is considered as Low level.

3.3.2 HEM for Smartphone Use Cases

Today, smartphones (as well as smart devices) provide numerous applications to the user based on their platform. To derive the importance of each metric component for the smartphone use case, we take the "next word prediction" as an example. Through literature review, we note that most next-word prediction applications do not reach very high accuracy (over 90%) and only support general prediction services [53, 54]. On the other hand, while service providers aim to deliver customized predictions, creating a personalized model for each individual client within a network is impractical due to the vast number of users. Thus, in the use case of "next word prediction," we rank the importance of Client Accuracy and Fairness as Moderate. As the environment changes dynamically with the user, smartphones usually suffer from fluctuating bandwidth and network. This unstable connection between client nodes and the server emphasizes the requirement of effectiveness Convergence of the FL model, with the importance level as High [54]. Furthermore, as most smartphones are designed with limited computational resources to facilitate portability, we classify the importance of Computational Efficiency as high-level.

3.3.3 HEM for Institution Use Cases

The institutional use cases include all organizations or institutions that are sensitive to data privacy and seek to benefit from FL systems. Here, we take healthcare institutions [45] as the example to derive and

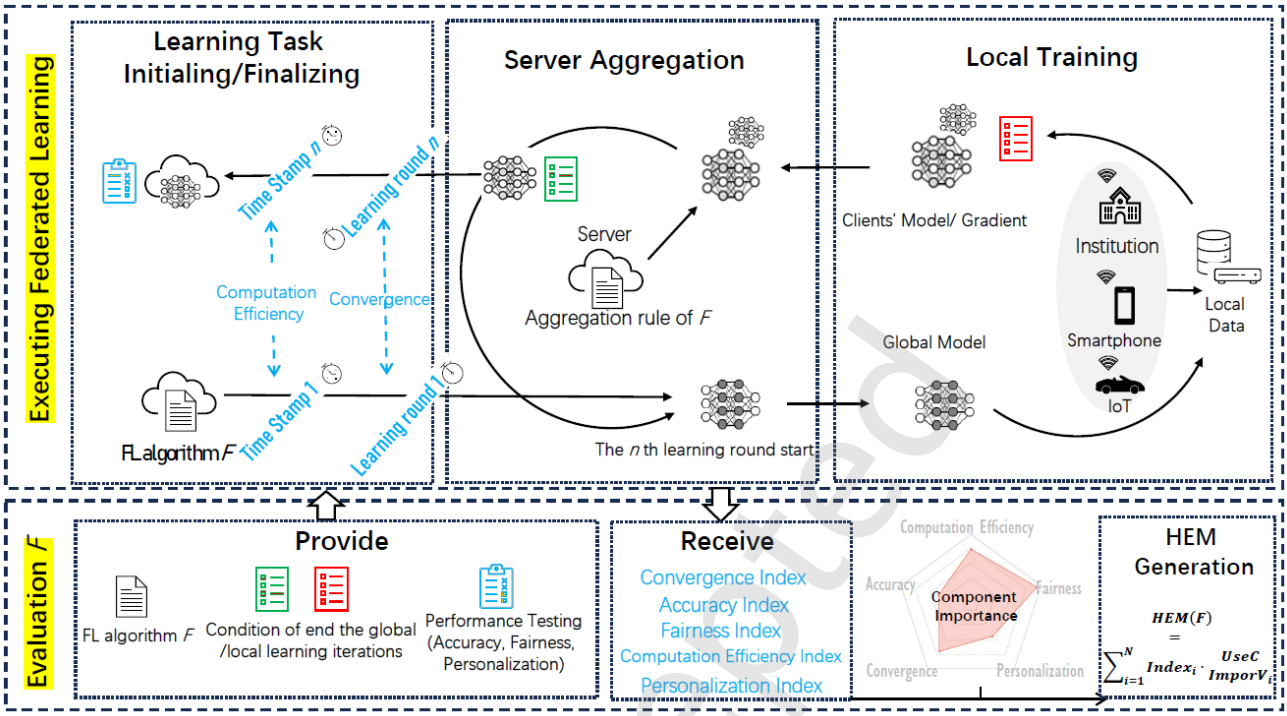


Fig. 1 The overall evaluation process with HEM for PFL in different use cases.

investigate the importance level for each component of HEM. Given that medical diagnoses contain highly sensitive patient information, individuals may not be willing to participate in the FL task and contribute their data if the algorithm does not offer a satisfactory service. Thus, Client Accuracy and Fairness achieve a High importance level in institutional use cases. On the other hand, institutions usually have stable networking, high bandwidth, and sufficient computational resources. They can afford computing-intensive learning tasks and achieve regular communication between the server and end nodes. Thus, the Convergence and Computational Efficiency are rated as low importance.

While we emphasize the importance of each component covering the most applications in three use cases, we acknowledge that certain specific applications within these categories may have unique requirements. Therefore, the importance levels indicated in Table 2 are intended as general guidelines, subject to further alignment based on the particular demands of individual applications.

3.4 Real-world Evaluation Process

In this section, we discuss the real-world evaluation process, from the preparation stage to the generation of the HEM results.

During the HEM evaluation procedure, the evaluator begins by selecting candidate FL algorithms, identifying learning tasks aligned with the specific use case, and defining the termination criteria for both local and global model training. These criteria include the target accuracy and a preset number of training rounds. Testing datasets are then collected to facilitate both global and local evaluations, ensuring that the testing data distribution appropriately reflects the non-IID characteristics of FL and the contextual requirements of the given use case.

When performing the evaluation, each selected FL algorithm undergoes standard training within the defined scenario, during which the training duration and the number of learning rounds are recorded until the predefined end conditions are met. To derive all evaluation components, the delivered models may be reinitialized and trained multiple times to satisfy various convergence thresholds. After training, the models are tested on the corresponding datasets to generate evaluation indices from multiple perspectives. These index values are then transformed into HEM scores based on the Importance Vector associated with each use case.

To enable real-world adaptability, the Importance Vector in HEM is not static but can be dynamically

updated according to evolving system objectives or user requirements. For instance, in practical FL deployments, the importance of metrics such as latency, fairness, or energy consumption may change over time due to environmental or policy variations. HEM allows these weights to be recalibrated periodically or automatically using feedback from system monitoring or performance logs, thereby maintaining its relevance under dynamic operational conditions. While Table 2 provides an initial guideline on component importance across different use cases, these importance levels can be further refined during system runtime to match the real-time priorities of the deployment environment. Ultimately, the resulting HEM scores support informed and adaptive selection of FL algorithms for continuous deployment and optimization. The overall HEM evaluation process is illustrated in Figure 1.

4 FL Algorithms Evaluation through HEM

In this section, we evaluate the most prominent FL algorithms and personalization methods using the proposed HEM index and the corresponding evaluation process. Initially, the evaluation focuses on each individual component of the metric. We then integrate these components into the HEM framework using the formula referenced in Equation (2). This section concludes with a discussion of the findings.

Algorithm	Accuracy	Convergence	Comp Eff	Fairness
FedAvg	0.84	0.67	0.12 (2.51s)	**1.00**
FedAvg_M	0.88	0.90	0.00 (3.64s)	0.55
FedAvg_P	0.88	0.85	0.78 (1.95s)	0.36
FedDyn	0.85	0.69	0.21 (1.68s)	0.41
FedDyn_M	**0.89**	**0.94**	0.56 (1.15s)	0.00
FedDyn_P	**0.89**	0.92	**0.89 (0.46s)**	0.27
SCAFF	0.86	0.86	0.44 (1.4s)	0.59
SCAFF_M	0.87	0.86	0.67 (0.95s)	0.23
SCAFF_P	**0.89**	0.91	0.87 (0.57s)	0.05

Table 3 The accuracy, convergence, computation efficiency, and fairness index (performances) of different FL algorithms.

4.1 Experiments Setup

We consider 100 clients participating in the FL tasks; each client trains the model with data on only 5 out of 10 classes in the chosen dataset. We set the end conditions as "1000 communication rounds" to generate the Client Accuracy index and "80% accuracy" to generate the Convergence index. We use the Cifar-10 as the training dataset, which consists of 60000, 32x32 pixels color images in 10 classes, including airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. Each algorithm was allotted 1 3070 GPU, and 32 GB of RAM. We perform the experiments five times and use the average as the final result.

As the overall data distribution in FL is considered non-IID, we introduce a custom dataset configuration to tailor the dataset to the requirements of our FL simulations [1]. Specifically, we distribute 5 classes of training and test data to each end node based on its class list. The ith client is assigned the class list shown as follows, where $\%$ denotes operation MOD:

$$(i+n)\%10, n \in [0,1,2,3,4] \tag{4}$$

We select the 5 most representative Fl algorithms for experiments, including FedAvg, FedDyn, SCAFFOLD, MAML (for personalized), and ProtoNet (for personalized). We use a CNN in our experiments, which has two convolutional layers (64× and 5×5 kernels), a max pooling layer, two fully connected layers (384×, 192×) with ReLU activation, and a softmax layer.

4.2 Evaluation Metric Component Indexes

Table 3 illustrates the Client Accuracy, Convergence, Computation Efficiency, and Fairness index (evaluation outcomes) of different FL algorithms and their personalized implementation. Figure 2 provides the radar plots of different FL algorithms.

4.2.1 Client Accuracy Index

The experimental results show that all FL algorithms achieve a high client model testing accuracy, with a variance in the accuracy index among them of less than 0.05. FedDyn, utilizing both MAML and Proto personalization methods, and SCAFFOLD, with Proto personalization, attain the highest performance, each achieving an accuracy of 0.89. In contrast, FedAvg witnesses the lowest testing accuracy at 0.84. The accuracy gap can be attributed to FedAvg's performance degradation when facing heterogeneous and non-IID data [10,44]. Furthermore, personalization methods such as MAML and Proto allow individual clients to receive FL models that are better fitted to their unique data distributions. Consequently, algorithms personalized through these methods show approximately 3% higher accuracy compared to their original algorithms.

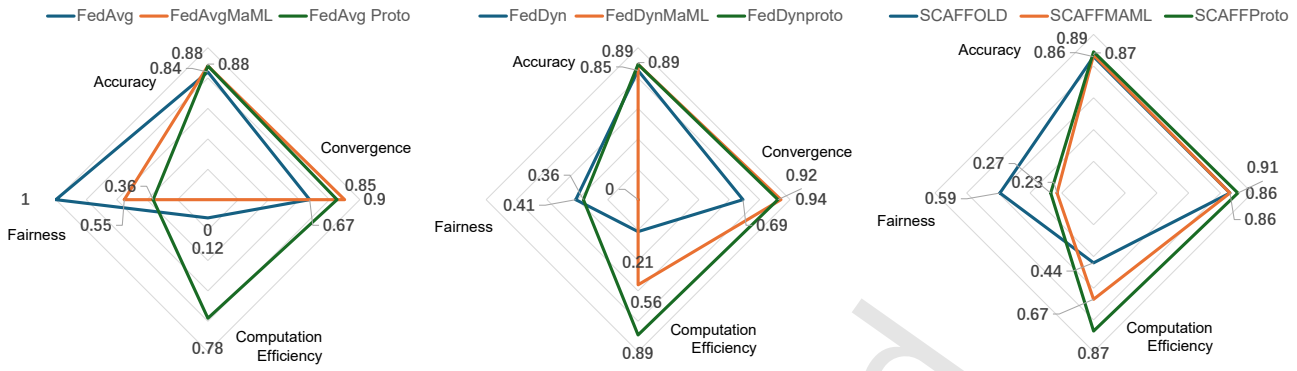


Fig. 2 Radar plots of FL algorithms: (left) FedAvg with variants, (middle) FedDyn with variants, (right) SCAFFOLD with variants.

4.2.2 Convergence Index

The Convergence Index across all simulated FL algorithms exhibits a wider range than the Client Accuracy, with the lowest performing algorithm, FedAvg, demonstrating a difference of approximately 0.17 when compared to the highest performing algorithm, FedDyn_MAML. Specifically, FedDyn_MAML, FedAvg_MAML, FedDyn_Proto, and SCAFFOLD_Proto show a higher performance, achieving a Convergence Index of over 0.90. In contrast, the lowest Convergence Index is observed in FedAvg and FedDyn, both falling below 0.7. This convergence performance difference is attributed to the accuracy gains in PFL algorithms achieved through the effectiveness of meta-learning techniques. Such techniques enable FL algorithms to transfer knowledge across clients, facilitating faster convergence [12]. In contrast, non-personalized FL algorithms may not consistently converge in limited communication rounds, particularly when trained on clients with strong non-IID data distribution (as simulated).

4.2.3 Computational Efficiency Index

The Computational Efficiency index is a comparative index across the simulated algorithms. As achieving the longest TTA, FedAvg_MAML is assigned to the lowest Computational Efficiency Index with 0 (3.6s). The experimental results indicate a strong correlation between the Computational Efficiency of FL algorithms and the personalization methods employed. Specifically, for each base FL algorithm, the incorporation of Proto personalization results in better computational efficiency compared to the integration of MAML. Besides, while recent computational efficiency research indicates FedAvg outperformed SCAFFOLD when the client datasets were IID [10], we note SCAFFOLD heavily outperformed FedAvg in non-IID scenarios. For instance, FedAvg, MAML, and Proto achieve 0.12 (2.51s), 0.00 (3.64s), and 0.78 (1.95s) Computation Efficiency Index, while SCAFFOLD, MAML, and Proto achieve 0.44 (1.4s), 0.67 (0.95s), and 0.87 (0.05s), respectively.

4.2.4 Fairness Index

The Fairness index serves as a comparative measure across the simulated algorithms, scaling from 0 to 1. Among the three non-personalized FL algorithms, FedAvg achieves the highest Fairness Index (indicating the lowest entropy) with a score of 1.00. It is followed by SCAFFOLD with a score of 0.59, FedAvg_MALA at 0.55, and FedDyn at 0.41. In contrast, FedDyn_MAML exhibits the highest entropy in its accuracy list, resulting in the lowest Fairness index of 0. Besides, the experimental results indicate that incorporating personalization methods into these FL algorithms increases the variance in clients' accuracy and leads to a reduction in the Fairness Index. Specifically, FedAvg, FedDyn, and SCAFFOLD exhibit reductions in their Fairness Index ranging from 0.45 to 0.64, 0.14 to 0.41, and 0.36 to 0.54, respectively.

4.2.5 Personalization of PFL Algorithms

The experimental results demonstrate that two personalization methods, Proto and MAML, effectively enhance model learning performance based on the client's specific data distribution, resulting in positive personalization indices for all evaluated algorithms. Specifically, FedAvg_Proto emerged as the most personalized FL algorithm, achieving a Median Percentage of Client Accuracy Improvement (MPI) of 10.46, closely followed by FedAvg_MAML.

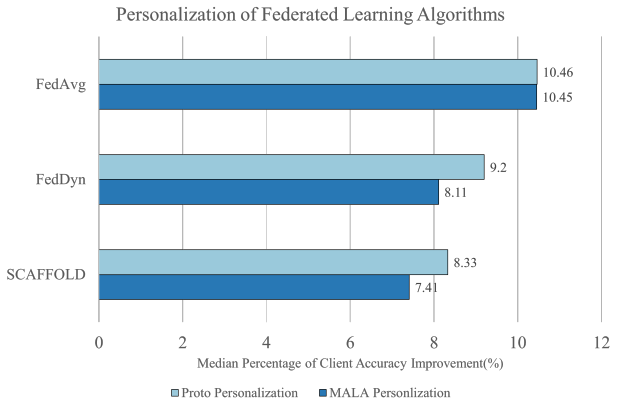


Fig. 3 Personalization of PFL algorithms

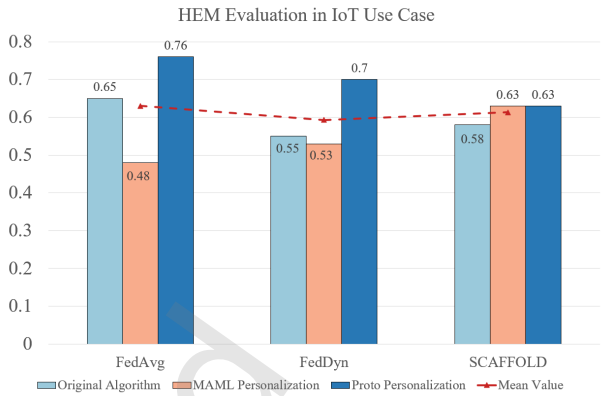


Fig. 4 HEM index of various FL algorithms in IoT use case.

The FedDyn class of PFL algorithms demonstrates moderate levels of personalization, receiving MPI values of 9.2 and 8.11 for the Proto and MAML methods, respectively. Conversely, SCAFFOLD PFL algorithms show the lowest degree of personalization, with an average MPI value of around 8.

Figure 3 illustrates the personalization capabilities of PFL algorithms.

4.3 Holistic Evaluation for the Three Use Cases

In this section, we conduct the HEM evaluation of the FL algorithms across the identified use cases (IoT, institution, and smartphone) based on the experimental results presented in Section 4.2 and Equation 3. The importance levels are quantified as follows: High is assigned an index of 3, Moderate is an index of 2, and Low is an index of 1 for organizing HEMs. Under this simulation, an FL algorithm exhibits Excellent overall performance if its HEM index is greater than 0.8. Performance is classified as Good for HEM indices ranging from 0.7 to 0.8, Acceptable for indices between 0.5 and 0.7, and Low for indices below 0.5. These quantifications and standards set here could be further updated according to the special requirements of target use cases.

4.3.1 IoT Use Case

In the IoT use case, the HEM index indicates that the FL algorithms' performance ranges from Good to Low. Specifically, FedAvg_Proto and FedDyn_Proto achieve the highest performance, with an HEM index of 0.76 and 0.70, respectively. FL algorithms based on SCAFFOLD receive the middle level with an average HEM index of around 0.60. In contrast, FedAvg_MAML and FedDyn_MAML receive the lowest HEM index, approximately 0.50. This variation in HEM performance is attributed to the High importance of Accuracy, Computational Efficiency, and Fairness in IoT use cases. Thus, FL algorithms that demonstrate high performance in these components (e.g., FedAvg_Proto) maintain an advantage in the HEM evaluation, whereas those with lower scores in these aspects (e.g., FedAvg_MAML) are at a disadvantage. Figure 4 illustrates the HEM index of various FL algorithms in IoT use cases.

Recall the individual competent performance evaluated shown in Table 3, the FedAvg_Proto algorithm does not reach the highest performance on each single perspective. As a result, a user seeking an FL algorithm for IoT use cases through traditional evaluation metrics may overlook the FedAvg_Proto, while it emerges as the most suitable algorithm when considering the IoT requirements. However, our proposed HEM reveals a different insight and shows the benefits of the FedAvg_Proto algorithm in the target scenarios. This comprehensive evaluation underscores the importance of considering the collective strengths of FedAvg_Proto rather than focusing on isolated metrics.

4.3.2 Smartphone Use Case

The HEM index shows that the performance of FL algorithms in the Smartphone use case ranges from Acceptable to Excellent. On the one hand, all three PFL algorithms that use the Proto meta-learning framework score in the Excellent Performance, achieving a HEM index of over 0.80. On the other hand, the original algorithms generally receive a low HEM index, with FedAvg at 0.56, FedDyn at 0.55, and SCAFFOLD at 0.64. From an average performance perspective, the SCAFFOLD class achieves the highest mean value

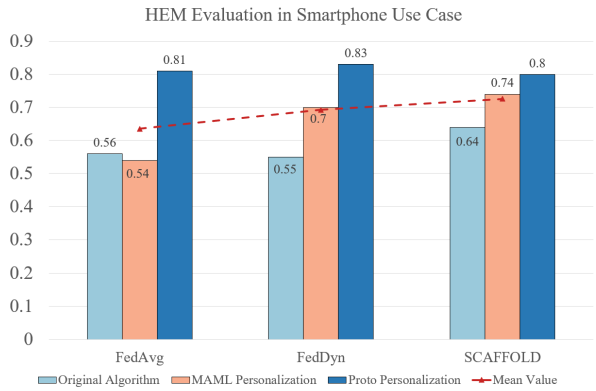


Fig. 5 HEM index of various FL algorithms in the Smartphone use case.

at 0.72, followed by FedDyn at 0.69 and FedAvg at 0.63, indicating its advantages in Smartphone use cases. Figure 5 illustrates the HEM index of various FL algorithms in the Smartphone use case. Given that

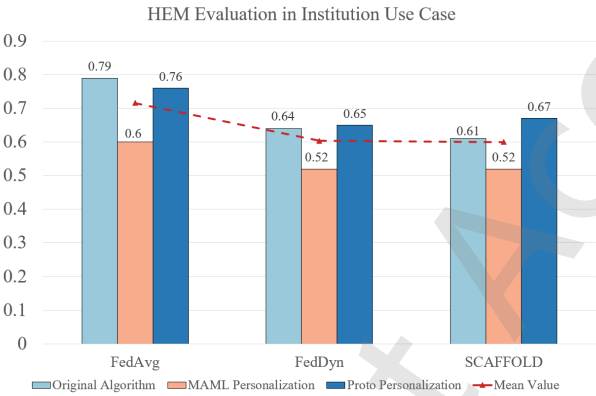


Fig. 6 HEM index of various FL algorithms in the Institution use case.

Convergence and Computational Efficiency are highly valued in the HEM evaluation for the smartphone use case, an FL algorithm that excels in these areas can compensate for deficiencies in other components. Thus, although FedDyn_Proto may not consistently achieve the highest performance in existing evaluations, it could be identified through the HEM as the most suitable algorithm for Smartphone users.

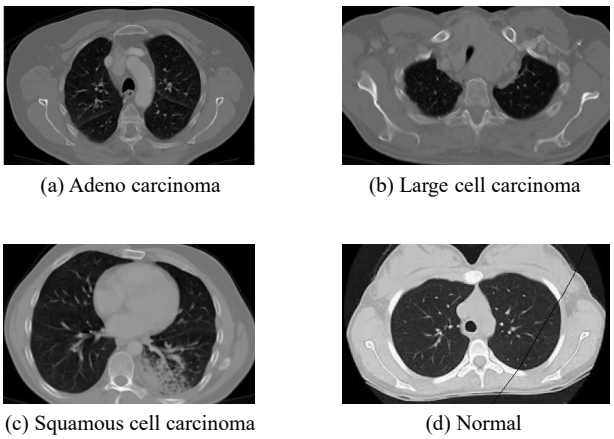
4.3.3 Institution Use Case

In the institution use case, the algorithms show performance from Good to Acceptable. Specifically, FedAvg scores the highest with an HEM index of 0.79, followed by FedAvg_Proto at 0.76 and FedDyn_Proto at 0.67. In contrast, FedDyn_MAML and SCAFFOLD achieve the lowest performance, which receives a 0.52 HEM index. As the Institution uses cases to prioritize Client Accuracy and Fairness, the HEM index for FedDyn_MAML is adversely impacted by its lower performance in fairness, despite achieving high model accuracy. Additionally, it is observed that most FL algorithms experience a decrease in their HEM index upon integrating MAML personalization, whereas the Proto method tends to maintain or even enhance performance. Figure 6 illustrates the HEM index of various FL algorithms in the simulated institution use case.

Based on the HEM evaluation results, a user seeking an FL algorithm to support model training and delivery services in institutional use cases might consider selecting FedAvg_Proto and FedDyn_Proto. Both algorithms deliver Good overall performance and satisfactory Fairness, which could continuously encourage clients to participate in the learning task.

4.4 Case Study: Chest Cancer Detection

To further demonstrate the effectiveness of HEM in real-world scenarios, we present a chest cancer classification case study. Following the experimental setup described in Section 4.1, we set the number of communication rounds to 20 and use VGG16 as the model for CPU-based training. In this case study, we consider a scenario where a healthcare authority aims to select an appropriate FL algorithm to deliver an intelligent chest cancer detection service. The training dataset includes three major chest cancer types, namely Adenocarcinoma, Large Cell Carcinoma, and Squamous Cell Carcinoma, along with normal cases, as illustrated in Figure 7.



(a) Adeno carcinoma (b) Large cell carcinoma

(c) Squamous cell carcinoma (d) Normal

Fig. 7 Illustration of the chest cancer classification data.

In this case study, we consider three FL candidate algorithms, namely FedAvg, FedDyn, and SCAFFOLD.

The healthcare use case has not been included among the three main scenarios discussed earlier. In this context, learning performance (accuracy) should be regarded as the important requirement (High). In addition, fairness should be given medium priority in healthcare applications to avoid bias in prediction outcomes, while the remaining components are considered of lower importance.

Table 4 presents the HEM index and the corresponding subindex values. It can be observed that all three candidate algorithms achieve comparable overall HEM scores. Among them, FedDyn demonstrates the best learning performance, achieving an accuracy of 0.78 and a convergence score of 0.94. In contrast, FedAvg exhibits superior fairness performance. From the perspective of HEM, FedAvg attains the highest overall index value of 1.36, and is therefore identified as the most suitable choice for this healthcare use case.

Algorithm	Accuracy	Conver	Com Eff	Fairness	HEM
FedAvg	0.77	0.73	0.42	1.00	1.36
FedDyn	0.78	0.94	0.53	0.66	1.28
SCAFF	0.73	0.82	0.74	0.69	1.28

Table 4 The learning performance of FL algorithms and their HEM index under the healthcare use case.

5 Discussion

The previous experimental results demonstrate that personalization methods enhance the Client Accuracy and Computation Efficiency of original FL algorithms by effectively converging on heterogeneous or non-IID data during the training process (see Table 3). However, it remains unclear whether these improvement carried by personalization methods implicitly brings trade-offs that could negatively affect other components and the overall HEM.

Thus, in this section, we seek to answer the following three research questions:

RQ1: What kind of trade-offs are carried by personalization methods?

RQ2: Whether the PFL algorithms with high personalization perform better?

RQ3: Whether the FL algorithms in each simulated use case benefited from the implementation of personalization methods?

Figure 8 illustrates the index fluctuation of various evaluation components by introducing personalization methods (MAML, Proto) to the FedDyn, SCAFFOLD, and FedAvg algorithms. One can observe that almost all components experience an increase following the introduction of personalization methods. Specifically, by introducing personalization, the Computational Efficiency achieves the largest improvement, with gains ranging from approximately 0.43 to 0.68 across different algorithms. This corresponds to a mean relative increase of about 110% compared with the baseline performance. Convergence and Accuracy exhibit smaller but consistent increases of around 0.22 and 0.03, respectively, equivalent to roughly 25% and 4% relative improvements. In contrast, Fairness shows a substantial decline, decreasing by 0.41 to 0.55, corresponding to an average relative drop of about 45%.

On the other hand, the Proto method is observed to impart a higher degree of personalization to the FL algorithm compared with the MAML method. As shown in Figure 3, Proto personalization yields higher MPI values than MAML by 0.01, 1.11, and 0.78 for FedAvg, FedDyn, and SCAFFOLD, respectively, corresponding to an average increase of 63% in personalization intensity. This stronger personalization leads to higher gains in Computational Efficiency, outperforming MAML by approximately 0.24 on average, which translates to an additional 35–40% efficiency improvement. However, this comes at the cost of reduced Fairness, which declines by about 0.2 compared with MAML, reflecting an average additional fairness degradation of 18%. Meanwhile, Accuracy and Convergence remain relatively stable, with fluctuations within ±0.03. These quantitative results confirm that higher personalization intensity brings diminishing returns in learning accuracy while amplifying fairness degradation. The observed numerical trends reinforce the inherent trade-off among efficiency, accuracy, and fairness, demonstrating the need for adaptive evaluation frameworks like HEM to balance these competing objectives under non-IID conditions.

From the perspective of use cases, we note that only two algorithms, FedAvg_Proto and FedDyn_Proto, both PFL algorithms, are rated as Good levels within the IoT scenario. Similarly, in both smartphone and institutional scenarios, algorithms achieving High and Good performance ratings are generally PFL, including FedAvg_Proto, FedDyn_Proto, and FedDyn_MAML. This observation indicates that despite the trade-offs associated with personalization methods, the

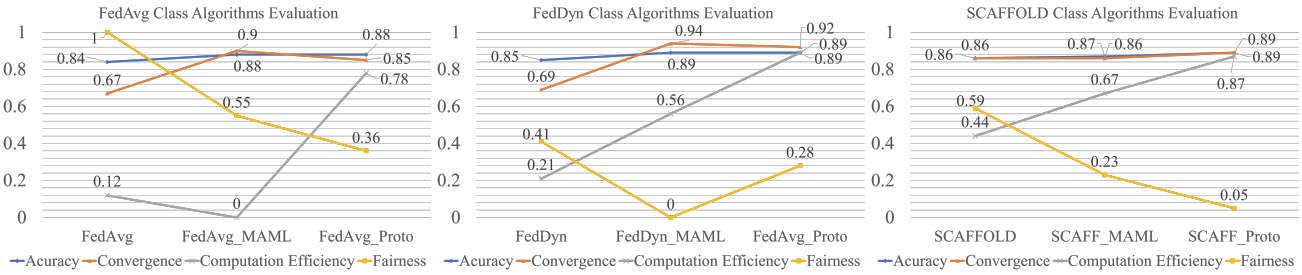


Fig. 8 Illustration of the index fluctuation in various evaluation components upon introducing personalization methods to the FedDyn, SCAFFOLD, and FedAvg algorithms.

overall HEM performance is enhanced, benefiting the application of FL algorithms across all three scenarios. As a result, PFL algorithms should be prioritized for selection within these simulated use cases.

However, since the generation of HEM is closely tied to the Importance Vector and its corresponding quantification, further discussion is warranted on whether improvements in convergence can offset decreases in fairness, ultimately contributing to a high HEM for a specific application. This consideration should be explored in conjunction with an investigation into the performance requirements of the deployment use case.

Limitations and Future Work: While the proposed HEM demonstrates strong effectiveness in evaluating federated learning (FL) algorithms and providing recommendations for various use cases, it currently has a limitation in the index aggregation process. Specifically, HEM relies on manually assigned weights designed by domain experts and a linear weighted aggregation strategy. This linear approach was adopted because it offers better interpretability and is easier to implement, allowing users to clearly understand the contribution of each evaluation metric to the overall score. However, such a design may not fully capture complex non-linear relationships and interactions among different evaluation dimensions, potentially limiting its adaptability to more intricate real-world scenarios.

To address these limitations, future work will explore automated approaches for determining the importance weights of each evaluation index, thereby improving adaptability and objectivity across different use cases. Moreover, since the current HEM framework is based on linear aggregation, we plan to investigate non-linear aggregation mechanisms that can adapt to diverse application scenarios and performance objectives, representing a promising direction for enhancing the robustness and expressiveness of the HEM framework.

6 Conclusion

In this paper, we outline our proposed HEM index and use three typical use cases to provide a comprehensive evaluation of the given FL algorithms in the respective scenarios. Specifically, the application scenarios of IoT, Smartphones, and institutions are included as the representative FL use cases. For each scenario, we identify the evaluation metric components and their corresponding importance vectors. The HEM index is generated through a combination of the evaluation metric components and importance vectors. We experimentally assess various FL and PFL algorithms through the HEM proposed in different use cases. The experimental results demonstrate that the HEM index can effectively and efficiently evaluate and select the appropriate FL algorithms for diverse scenarios.

References

[1] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, "A survey on federated learning: challenges and applications," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 513–535, 2023.

[2] Z. Hu, D. Li, K. Yang, Y. Xu, and B. Peng, "Optimizing data distributions based on jensen-shannon divergence for federated learning," *Tsinghua Science and Technology*, vol. 30, no. 2, pp. 670–681, 2024.

[3] D. Gufran and S. Pasricha, "Fedhil: Heterogeneity resilient federated learning for robust indoor localization with mobile devices," *ACM Transactions on Embedded Computing Systems*, vol. 22, no. 5s, pp. 1–24, 2023.

[4] P. Guo, P. Wang, J. Zhou, S. Jiang, and V. M. Patel, "Multi-institutional collaborations for improving deep learning-based magnetic resonance image reconstruction using federated learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2423–2432.

[5] M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R. R. Colen *et al.*, "Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data," *Scientific reports*, vol. 10, no. 1, pp. 1–12, 2020.

[6] J. Bian and J. Xu, "Client clustering for energy-efficient clustered federated learning in wireless networks," in *Adjunct Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp) & the ACM International Symposium on Wearable Computing (ISWC)*, 2023, pp. 718–723.

[7] Y. Li, W. Ding, H. Chen, W. Bao, and D. Yuan, "Contribution-wise byzantine-robust aggregation for class-balanced federated learning," *Information Sciences*, vol. 667, p. 120475, 2024.

[8] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[9] D. A. E. Acar, Y. Zhao, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," in *International Conference on Learning Representations (ICLR)*, 2021, pp. 1–36.

[10] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International Conference on Machine Learning (ICML)*, 2020, pp. 5132–5143.

[11] R. Liu, S. Yu, L. Lan, J. Wang, K. Kant, and N. Calleja, "A remedy for heterogeneous data: Clustered federated learning with gradient trajectory," *Big Data Mining and Analytics*, vol. 7, no. 4, pp. 1050–1064, 2024.

[12] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning: A meta-learning approach," *arXiv preprint arXiv:2002.07948*, 2020.

[13] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, "A review of applications in federated learning," *Computers & Industrial Engineering*, vol. 149, pp. 1–15, 2020.

[14] J. Mills, J. Hu, and G. Min, "Multi-task federated learning for personalised deep neural networks in edge computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 630–641, 2021.

[15] M. Wang, L. Zhou, X. Huang, and W. Zheng, "Towards federated learning driving technology for privacy-preserving micro-expression recognition," *Tsinghua Science and Technology*, vol. 30, no. 5, pp. 2169–2183, 2025.

[16] J. Ibrahim, Y. Li, H. Chen, and D. Yuan, "Holistic evaluation metrics for federated learning," in *International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 2024, pp. 1–6.

[17] N. Tabassum, M. Ahmed, N. J. Shorna, U. R. Sowad, M. Mejbah, and H. Haque, "Depression detection through smartphone sensing: A federated learning approach." *International Journal of Interactive Mobile Technologies*, vol. 17, no. 1, pp. 1–17, 2023.

[18] I. Varlamis, C. Sardianos, C. Chronis, G. Dimitrakopoulos, Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "Using big data and federated learning for generating energy efficiency recommendations," *International Journal of Data Science and Analytics*, vol. 16, no. 3, pp. 353–369, 2023.

[19] J. Ogier du Terrail, A. Leopold, C. Joly, C. Béguier, M. Andreux, C. Maussion, B. Schmauch, E. W. Tramel, E. Bendjebbar, M. Zaslavskiy *et al.*, "Federated learning for predicting histological response to neoadjuvant chemotherapy in triple-negative breast cancer," *Nature medicine*, vol. 29, no. 1, pp. 135–146, 2023.

[20] S. K. Lo, Y. Liu, Q. Lu, C. Wang, X. Xu, H.-Y. Paik, and L. Zhu, "Toward trustworthy ai: Blockchain-based architecture design for accountability and fairness of federated learning systems," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3276–3284, 2022.

[21] N. A. Jalali and H. Chen, "Federated learning security and privacy-preserving algorithm and experiments research under internet of things critical infrastructure," *Tsinghua Science and Technology*, vol. 29, no. 2, pp. 400–414, 2023.

[22] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and A. S. Avestimehr, "Federated learning for the internet of things: Applications, challenges, and opportunities," *IEEE Internet of Things Magazine*, vol. 5, no. 1, pp. 24–29, 2022.

[23] D. Wu, R. Ullah, P. Harvey, P. Kilpatrick, I. Spence, and B. Varghese, "Fedadapt: Adaptive offloading for iot devices in federated learning," *IEEE Internet of Things Journal*, vol. 9, no. 21, pp. 20 889–20 901, 2022.

[24] I. Dayan, H. R. Roth, A. Zhong, A. Harouni, A. Gentili, A. Z. Abidin, A. Liu, A. B. Costa, B. J. Wood, C.-S. Tsai *et al.*, "Federated learning for predicting clinical outcomes in patients with covid-19," *Nature medicine*, vol. 27, no. 10, pp. 1735–1743, 2021.

[25] Y. Wang, I. L. Bennani, X. Liu, M. Sun, and Y. Zhou, "Electricity consumer characteristics identification: A federated learning approach," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3637–3647, 2021.

[26] K. Sultana, K. Ahmed, B. Gu, and H. Wang, "Elastic optimization for stragglers in edge federated learning," *Big Data Mining and Analytics*, vol. 6, no. 4, pp. 404–420, 2023.

[27] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[28] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.

[29] D. A. E. Acar, Y. Zhao, R. Zhu, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama, "Debiasing model updates for improving personalized federated training," in *International Conference on Machine Learning (ICML)*, 2021, pp. 21–31.

[30] A. Xiong, H. Zhou, Y. Song, D. Wang, X. Wei, D. Li, and B. Gao, "A multi-task based clustering personalized federated learning method," *Big Data Mining and Analytics*, vol. 7, no. 4, pp. 1017–1030, 2024.

[31] X. Qu, C. Guan, G. Xie, Z. Tian, K. Sood, C. Sun, and L. Cui, "Personalized federated learning for heterogeneous residential load forecasting," *Big Data Mining and Analytics*, vol. 6, no. 4, pp. 421–432, 2023.

[32] M. Shayan, C. Fung, C. J. Yoon, and I. Beschastnikh, "Biscotti: A blockchain system for private and secure federated learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 7, pp. 1513–1525, 2020.

[33] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained iot devices," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 1–24, 2021.

[34] E. Farooq and A. Borghesi, "Lstm-based unsupervised anomaly detection in high-performance computing: A federated learning approach," in *2024 IEEE International Conference on Big Data (BigData)*. IEEE, 2024, pp. 7735–7744.

[35] J. Xu, H. Fan, Q. Wang, Y. Jiang, and Q. Duan, "Adaptive idle model fusion in hierarchical federated learning for unbalanced edge regions," *IEEE Transactions on Network Science and Engineering*, vol. 11, no. 5, pp. 4603–4616, 2024.

[36] S. A. Khowaja, K. Dev, P. Khowaja, and P. Bellavista, "Toward energy-efficient distributed federated learning for 6g networks," *IEEE Wireless Communications*, vol. 28, no. 6, pp. 34–40, 2022.

[37] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and Robust Federated Learning Through Personalization," Tech. Rep., 2021.

[38] H. Chen, T. Zhu, W. Zhou, and W. Zhao, "Afed: Algorithmic fair federated learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

[39] D. Chai, L. Wang, L. Yang, J. Zhang, K. Chen, and Q. Yang, "A survey for federated learning evaluations: Goals and measures," *IEEE transactions on knowledge and data engineering*, vol. 36, no. 10, pp. 5007–5024, 2024.

[40] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, "Heterogeneous federated learning: State-of-the-art and research challenges," *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–44, 2023.

[41] Y. Li, Z. Guo, N. Yang, H. Chen, D. Yuan, and W. Ding, "Threats and defenses in the federated learning life cycle: A comprehensive survey and challenges," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 9, pp. 15 643–15 663, 2025.

[42] S. Pandya, G. Srivastava, R. Jhaveri, M. R. Babu, S. Bhattacharya, P. K. R. Maddikunta, S. Mastorakis, M. J. Piran, and T. R. Gadekallu, "Federated learning for smart cities: A comprehensive survey," *Sustainable Energy Technologies and Assessments*, vol. 55, pp. 1–13, 2023.

[43] C. Yang, Q. Wang, M. Xu, Z. Chen, K. Bian, Y. Liu, and X. Liu, "Characterizing impacts of heterogeneity in federated learning upon large-scale smartphone data," in *Proceedings of the Web Conference (WWW)*, 2021, pp. 935–946.

[44] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE signal processing magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[45] M. Joshi, A. Pal, and M. Sankarasubbu, "Federated learning for healthcare domain-pipeline, applications and challenges," *ACM Transactions on Computing for Healthcare*, vol. 3, no. 4, pp. 1–36, 2022.

[46] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "Fedhealth: A federated transfer learning framework for wearable healthcare," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 83–93, 2020.

[47] Y. Li, X. Tao, X. Zhang, J. Liu, and J. Xu, "Privacy-preserved federated learning for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 8423–8434, 2021.

[48] D. C. Nguyen, Q.-V. Pham, P. N. Pathirana, M. Ding, A. Seneviratne, Z. Lin, O. Dobre, and W.-J. Hwang, "Federated learning for smart healthcare: A survey," *ACM Computing Surveys*, vol. 55, no. 3, pp. 1–37, 2022.

[49] X. Wang, W. Liu, H. Lin, J. Hu, K. Kaur, and M. S. Hossain, "Ai-empowered trajectory anomaly detection for intelligent transportation systems: A hierarchical federated learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 4, pp. 4631–4640, 2022.

[50] W. Zhang, D. Yang, W. Wu, H. Peng, N. Zhang, H. Zhang, and X. Shen, "Optimizing federated learning in distributed industrial iot: A multi-agent approach," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3688–3703, 2021.

[51] Y. Li, D. Yuan, A. S. Sani, and W. Bao, "Enhancing federated learning robustness in adversarial environment through clustering non-iid features," *Computers & Security*, vol. 132, p. 103319, 2023.

[52] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in iot," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5986–5994, 2019.

[53] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: State of the art, current trends and challenges," *Multimedia tools and applications*, vol. 82, no. 3, pp. 3713–3744, 2023.

[54] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heintz, and D. Roth, "Recent advances in natural language

processing via large pre-trained language models: A survey," *ACM Computing Surveys*, vol. 56, no. 2, pp. 1–40, 2023.

Yanli Li received the B.S. degree from Dalian University of Foreign Languages, Dalian, China, in 2017, the M.S. degree from University of Technology Sydney, Sydney, Australia, in 2020, and the Ph.D. degree from the University of Sydney, Sydney, Australia, in 2024. His primary research interests include federated learning, distributed computing, and adversarial machine learning.

Jehad Ibrahim is an undergraduate student at the University of Sydney, Sydney, NSW, Australia. With a strong passion for technology, he has focused his studies on Machine Learning, IoT, Software Engineering, and Telecommunications. Jehad's was awarded the Engineering Sydney Industry Placement Scholarship (ESIPS)

Huaming Chen is a Senior Lecturer with the School of Electrical and Computer Engineering, The University of Sydney, Sydney, NSW, Australia. He received the bachelor's and master's degrees from Lanzhou University, China, and the Ph.D. degree from the University of Wollongong, Australia. His research areas are trustworthy machine learning, software engineering and security, and machine learning application. He actively serves on the Program Committees and reviewers for top international journals and conferences, such as ACM CCS, ACM MM, IJCAI, KDD, SIAM ICDM, ECML/PKDD, ISSRE, ISSTA, ICSE and so on.

Dong Yuan received the B.Eng. and M.Eng. degrees from Shandong University, Jinan, China, in 2005 and 2008, respectively, and the Ph.D. degree from the Swinburne University of Technology, Melbourne, VIC, Australia, in 2012, all in computer science. He is currently an Associate Professor with the School of Electrical and Computer Engineering, The University of Sydney, Sydney, NSW, Australia. His research interests include machine learning, edge and cloud computing, and parallel and distributed systems.

Kim-Kwang Raymond Choo received his Ph.D. in Information Security from Queensland University of Technology, Australia, in 2006. He currently holds the Cloud Technology Endowed Professorship at The University of Texas at San Antonio, TX, USA. He is the founding co-EiC of ACM Distributed Ledger Technologies: Research and Practice and the founding chair of IEEE TEMS Technical Committee on Blockchain and Distributed Ledger Technologies. His research interests include cyber analytics, security and forensics.