

BlendFL: Blended Federated Learning for Handling Multimodal Data Heterogeneity

Alejandro Guerra-Manzanares[†], Omar El-Herroui[†], Michail Maniatakos and Farah E. Shamout

Division of Engineering

New York University Abu Dhabi, Abu Dhabi, UAE

Email: {alejandro.guerra, oe2015, michail.maniatakos, farah.shamout}@nyu.edu

Abstract—One of the key challenges of collaborative machine learning, without data sharing, is multimodal data heterogeneity in real-world settings. While Federated Learning (FL) enables model training across multiple clients, existing frameworks, such as horizontal and vertical FL, are only effective in ‘ideal’ settings that meet specific assumptions. Hence, they struggle to address scenarios where neither all modalities nor all samples are represented across the participating clients. To address this gap, we propose BlendFL, a novel FL framework that seamlessly blends the principles of horizontal and vertical FL in a synchronized and non-restrictive fashion despite the asymmetry across clients. Specifically, any client within BlendFL can benefit from either of the approaches, or both simultaneously, according to its available dataset. In addition, BlendFL features a decentralized inference mechanism, empowering clients to run collaboratively trained local models using available local data, thereby reducing latency and reliance on central servers for inference. We also introduce BlendAvg, an adaptive global model aggregation strategy that prioritizes collaborative model updates based on each client’s performance. We trained and evaluated BlendFL and other state-of-the-art baselines on three classification tasks using a large-scale real-world multimodal medical dataset and a popular multimodal benchmark. Our results highlight BlendFL’s superior performance for both multimodal and unimodal classification. Ablation studies demonstrate BlendFL’s faster convergence compared to traditional approaches, accelerating collaborative learning. Overall, in our study we highlight the potential of BlendFL for handling multimodal data heterogeneity for collaborative learning in real-world settings where data privacy is crucial, such as in healthcare and finance.

Index Terms—hybrid federated learning, multimodal learning, collaborative learning, decentralized inference, privacy-preserving machine learning

I. INTRODUCTION

Healthcare institutions collect a variety of vast heterogeneous medical data [1]. The heterogeneity of the data, like in other domains, stems from the fact that each institution, also referred to as a client, collects different data modalities from a specific set of users that may or may not be represented at other clients [2]. For example, one client may collect medical images while another collects laboratory test results. Hence, the overall aggregate dataset is considered to be multimodal, consisting of images, text, and/or numerical data, and heterogeneous due to varying levels of sparsity across the clients. Leveraging this aggregate dataset via collaborative learning to train centralized machine learning models could lead to

improved performance, such as for enhanced diagnostics [3]. However, in practice this is a very challenging task, not only due to privacy concerns [4], but also due to the nature of the data.

In this scenario, Federated Learning (FL) can be used to collaboratively train machine learning models without sharing sensitive patient data. FL frameworks are particularly important in settings where data sharing is restricted due to privacy and security reasons, such as healthcare institutions or financial organizations [4]. However, traditional FL frameworks like Horizontal Federated Learning (HFL) and Vertical Federated Learning (VFL) face challenges when applied in real-world scenarios where data can be asymmetrically fragmented and distributed unevenly across the clients [5]. HFL allows collaborative model training for clients that possess datasets with the same features but different data samples. Federated Averaging (FedAvg), proposed by [6] is the most common form of HFL [7]. In contrast, VFL deals with scenarios where clients hold different feature sets for the same data samples [8]. While both frameworks may be effective in settings where clients are only allowed to participate if they meet specific conditions, they struggle to address hybrid or ill-defined scenarios where neither all features nor all samples are available across all the clients [9]. This leads to suboptimal model training and inference capabilities, ultimately impeding non-conforming clients from participating in the federated network.

To address this gap and advance the applicability of FL in complex, real-world environments, we introduce BlendFL — a novel framework that seamlessly integrates the full capabilities of HFL and VFL, addressing their inherent incompatibility in a unique framework. BlendFL is designed to handle different types of data fragmentation, enabling the training of collaborative models on both horizontally partitioned data, where distinct clients contribute different samples with a consistent feature set, and vertically partitioned data, where shared samples across clients are characterized by different feature sets. This dual capability allows clients to participate in the collaborative framework and benefit from HFL, VFL, or both, regardless of their share of features and samples. By accommodating varying data fragmentation distributions across clients (e.g., partial, paired, fragmented), BlendFL enhances model robustness and applicability in federated learning contexts, ensuring effective utilization of diverse data sources.

In summary, we make the following contributions:

[†] These authors contributed equally to this work.

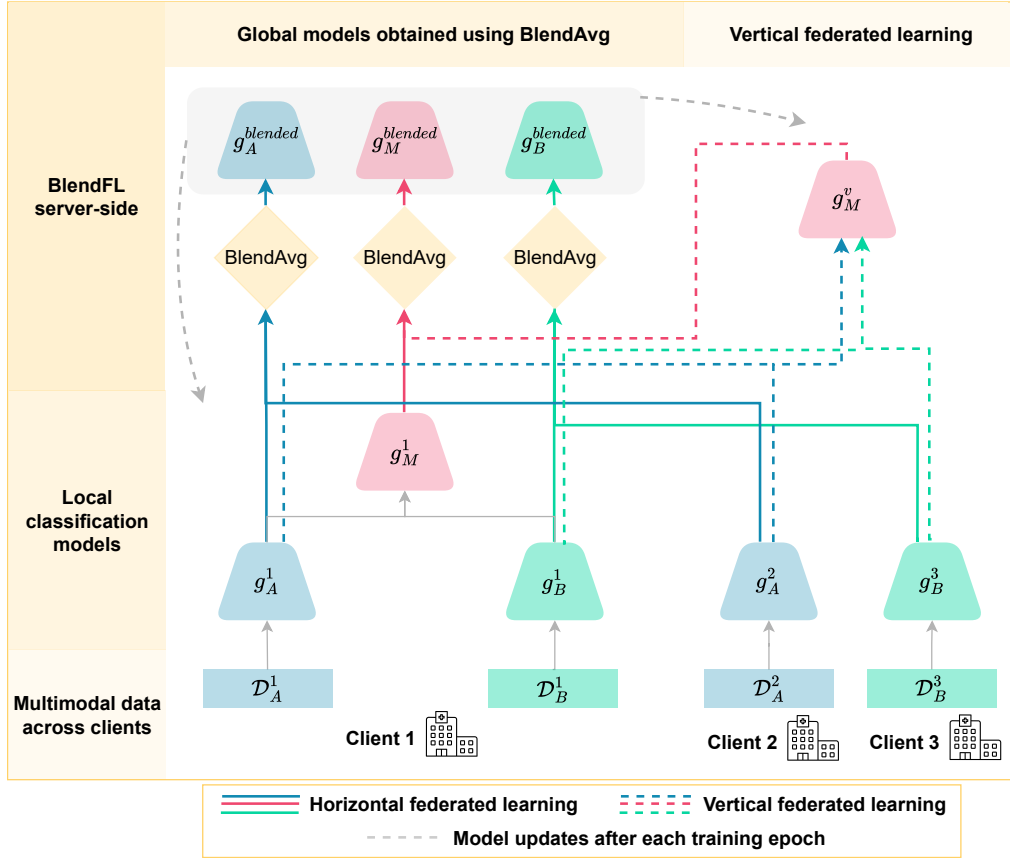


Fig. 1. **BlendFL architecture.** We showcase the BlendFL architecture with 3 hospitals: 1 has multimodal data (client 1), while the other 2 hospitals have only unimodal data (clients 2 and 3). Each party i has a specific dataset ($\mathcal{D}_m^i$), for a specific modality m , where $m \in \{A, B\}$. Note that data can be fragmented (different modalities for the same patient are in different hospitals), partial (only one modality exists for a patient), or paired (both modalities are present in the same client). The dashed and dotted arrows indicate communication between models, either locally or between the hospitals and the server, as detailed in the legend. For clarity, we omitted f_A and f_B , which are the feature extractors corresponding to each g_A and g_B classifier. Note that the arrows represent the data flow in one direction, to train the global models $g_A^{blended}$, $g_B^{blended}$, and $g_M^{blended}$. The grey dashed arrows represent the distribution of the global models back to the clients after every epoch.

- We propose **BlendFL**, a novel FL framework that seamlessly integrates HFL and VFL for collaborative training on unevenly distributed, heterogeneous datasets. This integration overcomes the limitations of standard FL paradigms, allowing clients to join the collaborative effort and benefit from HFL, VFL, or both. An overview is shown in Figure 1.
- Unlike conventional VFL settings, BlendFL enables clients to perform independent multimodal and unimodal predictions after training, which we present as **decentralized inference**. This reduces dependency on a central server for inference tasks and minimizes communication overhead, which is especially beneficial in scenarios like healthcare where timely and local decision-making is critical.
- We introduce **BlendAvg**, a novel weighted parameter averaging scheme for model updates. Unlike traditional federated averaging, BlendAvg determines weights based on the performance of each local model on a represen-

tative validation set, rather than based on the volume of data each client has. This ensures that the best-performing models have a greater influence on the global model, promoting updates only if the validation performance improves, thereby preventing model degradation due to overfitting.

- We perform **extensive validation** of BlendFL on two datasets across three multimodal tasks, including a real-world clinical dataset that simulates a realistic scenario where multiple hospitals collaborate to predict patient outcomes without sharing private data, validating its practicality in a realistic setting. BlendFL outperforms seven state-of-the-art baselines across all tasks.

II. RELATED WORK

Recent studies in FL have focused on either HFL or VFL [10]. In HFL, all clients hold datasets with identical features from different samples. One of the first HFL frameworks, FedAvg [6], utilizes a computation-then-aggregation strategy

to merge local model updates from various clients to form a global model. Since then, a wide variety of enhancement strategies have been proposed, such as SCAFFOLD [11], which shares additional data with the server to improve model convergence, and FedMA [12], which allows global and local models to be of different sizes. Note that all HFL frameworks assume that all client data have the same size and format [13].

In contrast, VFL deals with clients that hold different feature sets for common samples. Even though this setting has been notably less explored than HFL, one significant work is the FedBCD algorithm [14], which allows clients to perform multiple local updates before each communication, reducing communication overhead. Another study introduced FDML [15], which enables parties to perform asynchronous updates while mitigating the impact of stale information. Finally, SplitNN [16], the most popular implementation of VFL, proposes the use of a cut layer that allows clients to train a partial network up to a specific layer before passing intermediate outputs to a server, enabling the completion of the training process without accessing raw data.

A third type of FL frameworks, the so-called hybrid approaches, attempt to combine HFL and VFL to address more complex data distributions across clients. [13] propose the Hybrid Federated Matched Averaging (HyFEM) algorithm, which aligns features between local and global models using block coordinate descent, enhancing both privacy and model performance. Similarly, the FedHD [17] framework addresses the hybrid data challenge by incorporating gradient tracking and local stochastic gradient descent updates, which facilitates efficient communication and model training under partial client participation. In general, these hybrid methods are enhanced extensions of HFL incorporating some VFL capabilities, rather than frameworks that effectively combine both paradigms. To the best of our knowledge, BlendFL is the first framework to fully integrate HFL and VFL without imposing constraints or restrictions on clients.

BlendFL introduces a unified framework which directly combines both major, and a priori incompatible, FL paradigms in a synchronized and coherent manner. Moreover, this seamless integration makes BlendFL proficient in handling complex, multimodal datasets, unlike other FL proposals, including hybrid frameworks, which focus exclusively on unimodal datasets (e.g., ModelNet40 [13], MNIST [17]). Specifically, BlendFL is designed to harness the full potential of multimodal data in a unique federated framework characterized by data fragmentation and client heterogeneity through a novel global model update strategy (BlendAvg). Furthermore, VFL frameworks do not support local inference [18], requiring continued inter-client communication and synchronization for computing predictions. In contrast, BlendFL supports the development of robust unimodal and multimodal models that enable each client to perform local inference independently.

III. METHODOLOGY

We first introduce the problem setting along with formal notation. For the sake of clarity, we define our problem setting

within the healthcare domain. However, the framework is applicable to any domain where data privacy is important, and raw data sharing is not a feasible option for collaborative learning.

A. Problem setting

We assume that there are N healthcare institutions, i.e., hospitals, $C = \{c^1, c^2, \dots, c^N\}$, that seek to collaboratively train a global model using heterogeneous data collected from a global set of T patients, $U = \{u^1, u^2, \dots, u^T\}$. The heterogeneity stems from the fact that patients have their data collected either at a single hospital or multiple hospitals, the data is multimodal (e.g., clinical imaging and electronic health records), and the hospitals are unable to share the raw private data of the clients directly with each other due to privacy reasons. For simplicity, we assume that each patient u^i has either one or two data modalities collected across the hospitals, i.e. x_A^i and/or x_B^i .

Each hospital has a local dataset $\mathcal{D}^i$ collected from three possible types of patients:

- 1) Data of patients with a *paired* set of modalities such that both x_A and x_B were collected at the same hospital, denoted as $\mathcal{D}_{paired(A,B)}$,
- 2) Data of patients with a *fragmented* set of modalities, such that only one of the two modalities was collected at the hospital, while the other was collected at another hospital, denoted as $\mathcal{D}_{fragmented(A)}$ and $\mathcal{D}_{fragmented(B)}$,
- 3) Data of patients with a *partial* set of modalities, such that only one modality was collected at the hospital, and the other modality was never collected otherwise, denoted as $\mathcal{D}_{partial(A)}$ and $\mathcal{D}_{partial(B)}$.

Hence, a given hospital $c^i \in C$ has $\mathcal{D}^i = \{\mathcal{D}_A^i, \mathcal{D}_B^i\}$, where

$$\mathcal{D}_A^i = \{\mathcal{D}_{paired(A)}^i, \mathcal{D}_{fragmented(A)}^i, \mathcal{D}_{partial(A)}^i\}, \quad (1)$$

$$\mathcal{D}_B^i = \{\mathcal{D}_{paired(B)}^i, \mathcal{D}_{fragmented(B)}^i, \mathcal{D}_{partial(B)}^i\}, \quad (2)$$

or, to reflect real-world scenarios, a combination of datasets collected from different kinds of patients, i.e. paired, fragmented and/or partial for specific data modalities A and B .

Depending on the local set of data available $\mathcal{D}^i$, hospital c^i also has a set of encoders and classifiers for the two modalities. For simplicity, we assume that all modality-specific encoders are uniform across hospitals, though the number of encoders each client has may vary depending on the data modalities available to them. The feature encoders, f_A and f_B , process the modalities individually, such that:

$$h_A = f_A^i(x_A), h_B = f_B^i(x_B), \quad (3)$$

where h_A and h_B are latent representations. The local unimodal and multimodal classifiers compute the predictions for a given task, g_A , g_B , and g_M , such that:

$$\hat{y}_A = g_A^i(h_A), \hat{y}_B = g_B^i(h_B), \hat{y}_M = g_M^i(h_A, h_B). \quad (4)$$

Note that, in this setting, unimodal predictions are computed for local samples with missing modalities at a given hospital

Algorithm 1 BlendFL Training Procedure

Require: $\mathcal{D}^k$, data partitions across clients $k \in \{1, \dots, N\}$

Require: learning rate η , number of epochs E , number of clients N

Ensure: Trained models $g_A^{blended}, g_B^{blended}, g_M^{blended}$

```
1: Initialize server and client models  $g_A^k, g_B^k, g_M^k, g_M^v$ 
2: for  $e = 1$  to  $E$  do
3:   for each client  $k$  in parallel do
4:      $x_A^k, y_A^k \leftarrow$  Extract partial data from  $\mathcal{D}_A$ 
5:      $x_B^k, y_B^k \leftarrow$  Extract partial data from  $\mathcal{D}_B$ 
6:      $g_A^k \leftarrow$  TrainLocalPartial( $x_A^k, y_A^k$ )
7:      $g_B^k \leftarrow$  TrainLocalPartial( $x_B^k, y_B^k$ )
8:   end for
9:   for each client  $k$  in parallel do
10:     $x_A^k, y_A^k \leftarrow$  Extract fragmented data from  $\mathcal{D}_A$ 
11:     $x_B^k, y_B^k \leftarrow$  Extract fragmented data from  $\mathcal{D}_B$ 
12:     $h_A^k \leftarrow$  ClientForwardPass( $x_A^k, y_A^k$ )
13:     $h_B^k \leftarrow$  ClientForwardPass( $x_B^k, y_B^k$ )
14:    SendFeaturesToServer( $h_A^k$ )
15:    SendFeaturesToServer( $h_B^k$ )
16:   end for
17:   ServerAggregateFeatures( $h_A, h_B$ )
18:   ServerForwardPass()
19:    $g_M^v \leftarrow$  ServerBackwardPass()
20:   ServerSendGradientsToClients()
21:   for each client  $k$  do
22:      $g_M^k \leftarrow$  ReceiveGradientsAndBackwardPass()
23:   end for
24:   for each client  $k$  in parallel do
25:     if client  $k$  has local paired data then
26:        $x_A, x_B, y \leftarrow$  Extract paired data from  $\mathcal{D}^k$ 
27:        $g_M^k \leftarrow$  TrainLocalPaired( $x_A, x_B, y$ )
28:     end if
29:   end for
30:   ClientsSendWeightsToAggregationServer()
31:    $g_A^{blended}, g_B^{blended}, g_M^{blended} \leftarrow$  ServerBlendAvg()
32:    $g_A^k, g_B^k, g_M^k \leftarrow$  LocalUpdate( $g_A^{blended}, g_B^{blended}, g_M^{blended}$ )
33: end for
```

(i.e., fragmented and partial datasets), and multimodal predictions are computed for local multimodal samples (i.e. paired dataset).

Next, we describe the local models (encoders and classifiers) at the hospitals, the global server and its components, and associated assumptions for the possible collaborative learning scenarios. The local models at the hospitals are as follows:

- If the hospital only has paired samples (multimodal data for the same patients), then they locally have f_A, f_B, g_A, g_B , and g_M .
- If a hospital only has fragmented and/or partial data for a given modality, then it would have f_A and g_A , or f_B and g_B , without loss of generality.

Hospitals use their locally available encoders and classification models, both unimodal and multimodal, to perform local

training with their available local datasets. Local encoder and classification models are used for local training and compute local predictions as described in Eq. 3 and Eq. 4.

Then, if any two hospitals have complementary data based on overlap amongst patients (fragmented data, where the modalities for a patient are available but split across hospitals), they can collaborate, as in VFL, through the BlendFL server that has a global classifier trained using the complementary features of the local encoders:

$$\hat{y}_M^v = g_M^v(h_{\text{fragmented}(A)}, h_{\text{fragmented}(B)}). \quad (5)$$

Note that, for this step, we assume that all the collaborating hospitals share a common private database that includes identifiers for all individuals in the sample space, or that they implement a privacy-preserving dataset alignment technique, such as Private Set Intersection [19], to match and pair data modalities for shared individuals prior to joint multimodal model training with g_M^v .

Finally, to leverage the global set of data available at the hospitals, the BlendFL server, as in HFL, collects the locally trained models from the hospitals and combines them to form global models (encoders and classifiers) by aggregating the weights of the local models after each training iteration, such that:

$$g_A^{blended} = \text{BlendAvg}(g_A^i), \quad (6)$$

$$g_B^{blended} = \text{BlendAvg}(g_B^i). \quad (7)$$

Note that this parameter aggregation step is performed using the BlendAvg strategy, described in the next section. The same procedure is applied for the multimodal models. However, in this case, both the locally trained multimodal models (g_M^i) and the collaborative trained model (g_M^v) are aggregated to obtain a blended classifier:

$$g_M^{blended} = \text{BlendAvg}(g_M^i, g_M^v), \quad (8)$$

which is considered to be the final multimodal global model. Similarly, $g_A^{blended}$ and $g_B^{blended}$ are considered to be the final unimodal global models.

After obtaining all global models, a training iteration is completed. Then, the server distributes all global models to the collaborating hospitals, which use them to update the weights of their respective local models. Specifically, $g_M^{blended}$ is used to update the weights of the global and local multimodal models, while $g_A^{blended}$ and $g_B^{blended}$ are used to update their parameters of all local unimodal models.

B. BlendAvg

The Blended Averaging (BlendAvg) strategy aggregates the models' parameters based on each model's local improvement in terms of predictive power, as measured by performance metrics on a validation set. This predictive performance serves as a reliability metric for the local model parameters per hospital and is used to regulate the impact of local model contributions to the global model update.

Consider the presence of L models that are locally trained and used for global model averaging, where the i -th model

has parameters indicated by W_i . In conventional HFL, the aggregation server uses the traditional FedAvg [6] strategy to combine the parameters of the local models and output an averaged model, where all models contribute equally. On the other hand, BlendAvg proposes to aggregate the models' parameters proportional to the predictive power of each model. First, it calculates a weighting coefficient for each model based on a predictive performance score, denoted as A_i . For all local models, A_i is calculated at the aggregation server using a private representative validation dataset that is randomly sampled from the collaborating clients. After evaluating the performance of each received model, the aggregation server proceeds with model aggregation as follows:

- 1) **Measurement of local training improvement.** First, it calculates the improvement in predictive performance on the validation set for each model (A_i) compared to the previous global model performance on the same validation set (A_{global}). This assesses whether the local model training improved or not based on:

$$\Delta_i = A_i - A_{global}, \quad (9)$$

where $\Delta_i > 0$ indicates improvement by the local training on the validation set with respect to A_{global} , while $\Delta_i \leq 0$ indicates that the local training did not provide any improvement on the validation set with respect to A_{global} . Note that A_{global} is calculated using the previous $g_M^{blended}$, on the same validation set, in the case of multimodal blending as defined in Eq. 8. For the unimodal encoders, A_{global} is calculated using the previous $g_A^{blended}$ and $g_B^{blended}$ for Eq. 6 and Eq. 7, respectively.

Note that this step and the subsequent steps are performed independently for each unimodal and multimodal models, as there is a global model for each one of them (see Figure 1). However, for the sake of understanding and to avoid redundancy, the following steps are only described for a single generic computation (e.g., $g_i^{blended}$, where i represents the i -th modality or multimodal data).

- 2) **Weight calculation and normalization.** The subset of l_d models, $l_d \in L$, reporting $\Delta_i \leq 0$ are discarded and not used for updating the global model. The subset of l_u models, $l_u \in L$, with associated improvements ($\Delta_i > 0$) are considered for the global model update. Note that $l_d + l_u = L$. Next, the weighting coefficients (ω_i) are calculated as follows:

$$\omega_i = \frac{\Delta_i}{\sum_{i=1}^{l_u} \Delta_i}. \quad (10)$$

Each model in l_u receives an associated weighting coefficient. Dividing by the sum of all improvements ensures that the sum of the weighting coefficients adds up to 1, assigning proportional weights to each model.

- 3) **Weighted averaging.** The final model parameters ($W_i^{blended}$) are calculated as the weighted sum of each l_u

local model parameters (W_i) multiplied by its weighting coefficient ω_i . Formally, it is defined as follows:

$$W_i^{blended} = \sum_{i=1}^{l_u} \omega_i \times W_i. \quad (11)$$

Following this procedure, $W_i^{blended}$ only incorporates the parameters of the best performing models for the model update. This blending strategy ensures that each model's contribution to the final aggregated global model is proportional to its performance improvements, fostering a more adaptive and performance-oriented global model. Unlike traditional methods that average model parameters based on static criteria or data volume, like FedAvg, this approach allows for a dynamic adaptation to changes in model performance over time.

C. Execution details

We depict an example of the BlendFL framework in Fig. 1, which illustrates a scenario where three healthcare institutions participate in the collaborative training effort. In this scenario, each hospital has a subset of patients (samples). Each patient can have two possible modalities (A and B), which define a feature space. For instance, modality A could be Chest X-ray imaging data, and modality B could be electronic health records. Each available data modality is associated with a unimodal encoder. If both modalities are present for some patients within the same hospital, then the hospital also has a multimodal model. Note that the arrows indicate the data flow from the hospital to the BlendFL server. However, for the sake of clarity, the updating of the local models based on the global models (from servers to clients) is not depicted in detail but follows the reverse path from the BlendFL server to the hospitals.

The BlendFL framework, as outlined in Algorithm 1, orchestrates the simultaneous training of horizontal and vertical federated learning across multiple clients with heterogeneous data. Note that, to make our framework generic and context-agnostic, we use the term 'client' in Algorithm 1, as it is customary in federated learning literature, which equates to 'hospital' in our descriptive example. Following Algorithm 1, at each training epoch, the available local datasets at the hospitals are used sequentially to train the models as follows:

- 1) **Local training with partial data.** Partial data is utilized to train local unimodal models at each hospital (lines 3-8 in Algorithm 1). If a hospital has only one modality then it would process that modality.
- 2) **Multimodal global model training with fragmented data.** Each hospital holding fragmented data for a subset of patients performs a forward pass to compute intermediate features, which are then sent to the BlendFL server (lines 9-16 in Algorithm 1). If a hospital has only one modality, it would therefore execute only the process for that modality. The BlendFL server aligns (pairs) the intermediate features received from the hospitals for the subset of patients with fragmented data and performs a forward and backward pass on the multimodal model

TABLE I
PERFORMANCE RESULTS OF BLENDFL, CENTRALIZED LEARNING, AND FEDERATED BASELINES FOR CLINICAL CONDITIONS PREDICTION. BEST RESULTS ARE SHOWN IN BOLD.

Method	Multimodal		EHR		CXR	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
Centralized	0.746 (0.733, 0.761)	0.395 (0.357, 0.434)	0.759 (0.742, 0.778)	0.420 (0.380, 0.461)	0.707 (0.689, 0.725)	0.408 (0.369, 0.448)
FedAvg [6]	0.713 (0.698, 0.729)	0.358 (0.319, 0.398)	0.748 (0.731, 0.766)	0.403 (0.363, 0.445)	0.677 (0.659, 0.696)	0.400 (0.362, 0.440)
FedMA [12]	0.722 (0.707, 0.737)	0.366 (0.327, 0.406)	0.742 (0.725, 0.759)	0.399 (0.360, 0.440)	0.692 (0.674, 0.711)	0.396 (0.357, 0.435)
FedProx [20]	0.712 (0.697, 0.728)	0.355 (0.316, 0.395)	0.747 (0.730, 0.765)	0.401 (0.361, 0.441)	0.693 (0.674, 0.712)	0.397 (0.358, 0.437)
FedNova [21]	0.705 (0.690, 0.721)	0.347 (0.307, 0.386)	0.746 (0.729, 0.763)	0.401 (0.361, 0.441)	0.676 (0.657, 0.696)	0.403 (0.364, 0.443)
One-Shot VFL [22]	0.711 (0.696, 0.726)	0.352 (0.313, 0.392)	0.742 (0.725, 0.760)	0.391 (0.352, 0.432)	0.681 (0.662, 0.701)	0.402 (0.363, 0.442)
HFCL [23]	0.698 (0.683, 0.714)	0.341 (0.302, 0.381)	0.734 (0.718, 0.752)	0.382 (0.343, 0.422)	0.684 (0.666, 0.703)	0.388 (0.349, 0.427)
SplitNN [16]	0.706 (0.690, 0.722)	0.341 (0.301, 0.381)	0.741 (0.723, 0.760)	0.391 (0.351, 0.432)	0.680 (0.662, 0.699)	0.398 (0.359, 0.437)
BlendFL	0.732 (0.717, 0.747)	0.375 (0.336, 0.415)	0.753 (0.735, 0.770)	0.408 (0.368, 0.448)	0.704 (0.686, 0.723)	0.430 (0.391, 0.469)

TABLE II
PERFORMANCE RESULTS OF BLENDFL, CENTRALIZED LEARNING, AND FEDERATED BASELINES FOR IN-HOSPITAL MORTALITY PREDICTION. BEST RESULTS ARE SHOWN IN BOLD.

Method	Multimodal		EHR		CXR	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
Centralized	0.866 (0.852, 0.880)	0.501 (0.460, 0.541)	0.867 (0.853, 0.881)	0.491 (0.451, 0.531)	0.729 (0.710, 0.748)	0.181 (0.143, 0.220)
FedAvg [6]	0.844 (0.830, 0.858)	0.432 (0.392, 0.472)	0.856 (0.841, 0.871)	0.498 (0.457, 0.538)	0.688 (0.668, 0.708)	0.167 (0.129, 0.205)
FedMA [12]	0.849 (0.835, 0.863)	0.496 (0.456, 0.536)	0.851 (0.836, 0.866)	0.508 (0.468, 0.548)	0.630 (0.610, 0.650)	0.141 (0.103, 0.179)
FedProx [20]	0.845 (0.831, 0.859)	0.484 (0.444, 0.524)	0.849 (0.834, 0.864)	0.514 (0.474, 0.554)	0.606 (0.586, 0.626)	0.127 (0.089, 0.165)
FedNova [21]	0.843 (0.829, 0.857)	0.477 (0.437, 0.517)	0.849 (0.834, 0.864)	0.515 (0.475, 0.555)	0.726 (0.708, 0.748)	0.188 (0.150, 0.226)
One-Shot VFL [22]	0.848 (0.834, 0.862)	0.495 (0.455, 0.535)	0.854 (0.839, 0.869)	0.507 (0.467, 0.547)	0.717 (0.697, 0.737)	0.189 (0.151, 0.227)
HFCL [23]	0.839 (0.825, 0.853)	0.471 (0.431, 0.511)	0.844 (0.829, 0.859)	0.506 (0.466, 0.546)	0.582 (0.562, 0.602)	0.125 (0.087, 0.163)
SplitNN [16]	0.825 (0.810, 0.840)	0.415 (0.375, 0.455)	0.849 (0.834, 0.864)	0.461 (0.421, 0.501)	0.703 (0.683, 0.723)	0.169 (0.131, 0.207)
BlendFL	0.865 (0.851, 0.879)	0.494 (0.453, 0.534)	0.864 (0.849, 0.879)	0.513 (0.472, 0.553)	0.727 (0.708, 0.746)	0.195 (0.157, 0.233)

(lines 17-19 in Algorithm 1). This completes a training epoch for g_M^v (hold at the server). The gradients generated during this backward pass are decoupled and sent back to the respective hospitals to complete a full training cycle (lines 20-23 in Algorithm 1).

- 3) **Local training with paired data.** Hospitals use their paired data (both modalities are present) to train the local multimodal model (lines 24-29 in Algorithm 1).

This concludes a local training epoch for all hospitals using all locally available data. Then, the process continues as follows (lines 30-32 in Algorithm 1):

- 1) Unimodal and multimodal model parameters from all hospitals are sent to the server for weight aggregation.
- 2) The server aggregates the model parameters using BlendAvg (see Section III-B) for both unimodal and multimodal models, as defined in Eq. 6, Eq. 7, and Eq. 8.
- 3) The parameters of the aggregated models (i.e., $g_A^{blended}$, $g_b^{blended}$, $g_M^{blended}$ in Fig. 1) are distributed to the hospitals, which update their local models, concluding a global training epoch.

This training process repeats until the pre-defined number of training epochs is reached. This iterative process results in one blended global multimodal model and one blended global unimodal model per modality. Through non-restrictive collaborative learning, seamlessly blending vertical and horizontal federated learning, the BlendFL framework leverages all data available at the clients (hospitals), regardless of their available local sample (patients) and feature (modalities) spaces.

IV. EXPERIMENTS

In this section, we first evaluate our proposed framework for the clinical setting depicted in Fig. 1 on two multimodal tasks using a widely-used real-world clinical dataset. Next, we evaluate the generalization of our proposal on another dataset and different multimodal architecture. We also conduct convergence experiments for our novel averaging method, BlendAvg, and perform ablation studies to assess the impact of data distribution and number of clients for BlendFL and relevant baselines. For reproducibility, we make our code publicly available at <https://github.com/nyuad-cai/BlendFL>.

A. Datasets and Tasks

For the clinical tasks, we use the **MIMIC-IV** [24] and **MIMIC-CXR** [25] datasets, which are real-world clinical datasets consisting of Electronic Health Record (EHR) data for over 65,000 patients admitted to an Intensive Care Unit (ICU) and 377,100 Chest X-Ray (CXR) images, respectively. We paired the imaging data with associated clinical time-series data. We follow the same data splits and clinical tasks used by [26], which are described as follows:

- **Clinical conditions prediction:** This multilabel classification task aims to predict a set of 25 different clinical conditions for each ICU stay. The task utilizes time-series data from the entire ICU record paired with the last CXR image collected during the same stay. The output is a vector of 25 binary phenotype labels, indicating the presence of one or more conditions for a given patient.

TABLE III
PERFORMANCE RESULTS OF BLENDFL, CENTRALIZED LEARNING, AND FEDERATED BASELINES FOR MULTIMODAL AND UNIMODAL PREDICTIONS ON THE S-MNIST DATASET. THE BEST COLLABORATIVE FRAMEWORK RESULTS ARE SHOWN IN BOLD.

Method	Multimodal		Audio		Image	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
Centralized	0.989 (0.983, 0.994)	0.928 (0.917, 0.940)	0.807 (0.781, 0.833)	0.409 (0.371, 0.446)	0.982 (0.974, 0.990)	0.912 (0.901, 0.923)
FedAvg [6]	0.956 (0.943, 0.969)	0.815 (0.798, 0.832)	0.716 (0.689, 0.743)	0.281 (0.241, 0.320)	0.963 (0.950, 0.976)	0.860 (0.846, 0.874)
FedMA [12]	0.857 (0.841, 0.873)	0.497 (0.456, 0.538)	0.659 (0.630, 0.688)	0.198 (0.159, 0.237)	0.945 (0.932, 0.958)	0.699 (0.664, 0.734)
FedProx [20]	0.953 (0.938, 0.968)	0.808 (0.778, 0.838)	0.724 (0.695, 0.753)	0.288 (0.247, 0.329)	0.962 (0.951, 0.975)	0.854 (0.828, 0.880)
FedNova [21]	0.957 (0.942, 0.972)	0.824 (0.794, 0.854)	0.722 (0.693, 0.751)	0.286 (0.245, 0.327)	0.963 (0.950, 0.976)	0.855 (0.829, 0.881)
One-Shot VFL [22]	0.829 (0.813, 0.845)	0.474 (0.433, 0.515)	0.630 (0.601, 0.659)	0.169 (0.130, 0.208)	0.916 (0.903, 0.929)	0.670 (0.635, 0.705)
HFCL [23]	0.936 (0.921, 0.951)	0.742 (0.712, 0.772)	0.685 (0.656, 0.714)	0.239 (0.200, 0.278)	0.945 (0.932, 0.958)	0.784 (0.751, 0.817)
SplitNN [16]	0.942 (0.928, 0.956)	0.776 (0.758, 0.794)	0.718 (0.690, 0.746)	0.273 (0.234, 0.311)	0.958 (0.948, 0.968)	0.827 (0.812, 0.842)
BlendFL	0.983 (0.977, 0.989)	0.914 (0.902, 0.926)	0.803 (0.777, 0.829)	0.412 (0.374, 0.450)	0.978 (0.969, 0.987)	0.893 (0.881, 0.905)

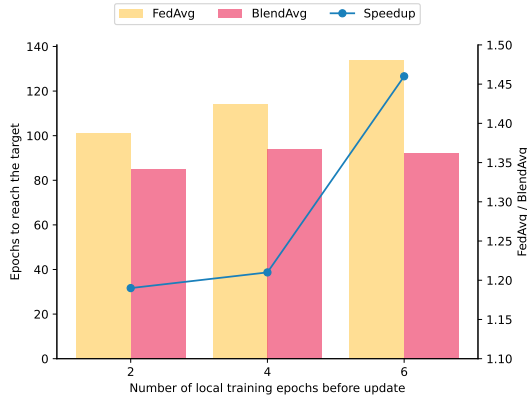


Fig. 2. **Model Convergence.** Comparison of rounds needed for model convergence (to reach the target 0.98 AUROC) for the federated model update strategies BlendAvg and FedAvg on the S-MNIST dataset.

We used a total of 42,636 EHR samples and 124,740 CXR samples. The data was divided into 70% training, 10% validation, and 20% test sets.

- **In-hospital mortality prediction:** This binary classification task predicts in-hospital mortality based on clinical data from the first 48 hours of an ICU stay. Only stays longer than 48 hours are considered, and each instance is paired with the last CXR image collected during the ICU stay. This task utilized 18,843 EHR samples and 124,740 CXR samples, divided using the same split as for the clinical conditions task.

To assess the generalization of our approach to other tasks, we used **S-MNIST** [27], which is an audio-visual dataset designed for benchmarking multimodal classification. It pairs the original MNIST dataset [28] with a spoken digits database from Google Speech Commands [29]. For our experiments, we randomly sample a subset of the original training set (500 instances). A smaller training dataset enabled us to simulate realistic scenarios, providing a more stringent test of each model’s ability to generalize from smaller, less comprehensive datasets. It also improved our analysis of model behavior under constrained data, common in federated learning. The validation and test sets both consist of 10,000 instances.

B. Model architectures

For both clinical tasks using MIMIC-IV and MIMIC-CXR (multilabel and binary classification), we used the architecture proposed by [26], which consists of an LSTM encoder as the EHR data feature extractor and a ResNet-34 [30] as the CXR image feature extractor. The features are then processed and fused to compute the final multilabel/binary prediction.

For the multiclass classification task with S-MNIST, we used a multimodal fusion architecture composed of two ResNet-18 [30] encoders as unimodal (audio and image) feature extractors. The unimodal features are concatenated and used as input for a linear layer that provides the final multiclass prediction.

C. Baselines

The proposed hybrid FL methodologies focus on improving specific aspects of HFL or VFL frameworks, such as synchronization [17] or parameter matching [13], by developing additional features for particular characteristics of unimodal data, without considering a holistic integration of HFL and VFL in a multimodal setting as BlendFL does. For that reason, more appropriate baselines for our proposed framework are the foundational FL frameworks such as horizontal federated learning, vertical federated learning, and also centralized learning. The baselines are briefly described as follows:

- **FedAvg** [6] is the foundational form of Federated Learning (FL), where each client trains a local model on its own dataset and periodically shares model updates with a central server. The server aggregates these updates to create a global model, which is sent back to the clients.
- **FedMA** [12] is an optimized form of HFL for CNN and LSTM architectures. It introduces *matched averaging*, which constructs the shared model by matching and aggregating hidden elements in a layer-wise fashion.
- **FedProx** [20] is an optimized form of HFL that introduces reparametrization and a proximal term to generalize FedAvg [6] to tackle heterogeneity in federated networks.
- **FedNova** [21] is a form of HFL that combines the FedProx and FedAvg approaches, focusing on addressing objective inconsistency and bias by proposing a novel weighting scheme for local models during averaging.

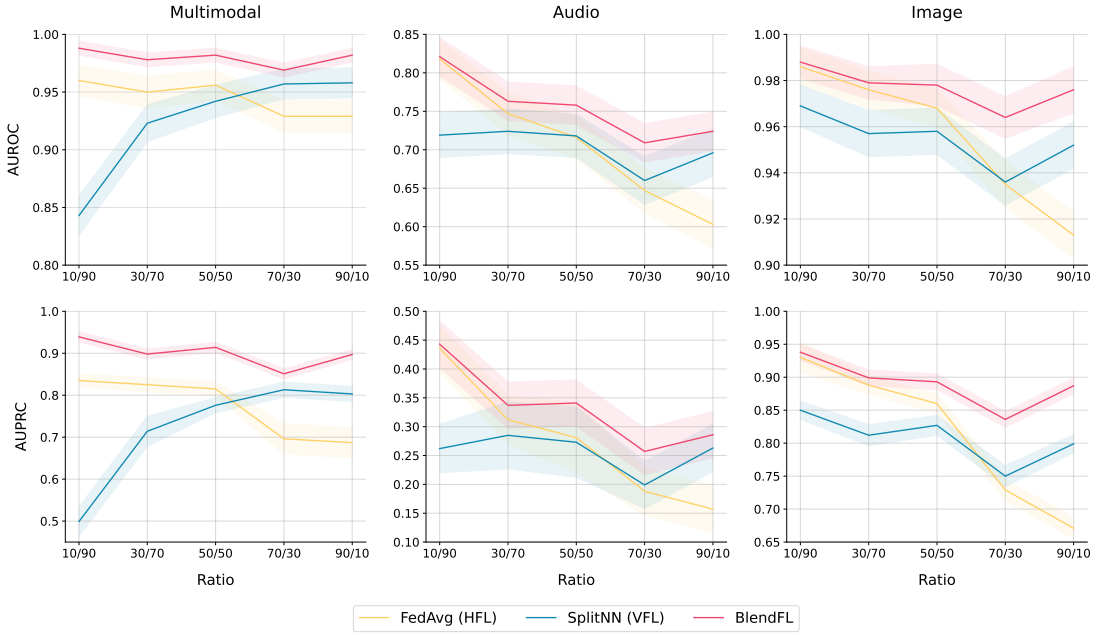


Fig. 3. **Data Distribution.** Performance comparison of BlendFL and FL baselines for different data distribution ratios (paired/partial) on the S-MNIST dataset.

- **HFCL [23]** is a hybrid FL framework where only clients with enough computational resources are involved in the FL training process. The remaining clients share their data with a central server that performs training on their behalf.
- **SplitNN [16]** is the most common implementation of VFL, where parties train a collaborative model in which different clients hold different subsets of features for the same samples
- **One-Shot VFL [22]** is a VFL framework that proposes local semi-supervised learning to address the problems of high communication cost and limited data overlap.
- **Centralized learning** is the traditional ML approach where all data is pooled into a single central server that performs model training. It assumes no privacy concerns and unrestricted data access. Although often infeasible in many real-world applications due to privacy issues, it serves as a strong baseline and performance upper bound for evaluating federated learning methods.

D. Convergence and Speedup Experiments

To evaluate the efficiency and effectiveness of our proposed BlendFL framework, we conducted convergence experiments to compare FedAvg with BlendAvg. These experiments are crucial for understanding the impact of aggregation strategies on the rate of convergence in federated settings. Methods that enable faster convergence reduce communication overhead, lower associated energy consumption, improve scalability, and enhance privacy preservation (due to less time of exposure to possible attacks). In our experimental setup, we measured the number of training rounds (epochs) required to reach a target

performance metric, i.e., AUROC of 0.98, using FedAvg and BlendAvg. We quantify the speedup ratio as:

$$\text{Speedup} = \frac{\# \text{ epochs to reach the target using FedAvg}}{\# \text{ epochs to reach the target using BlendAvg}}$$

E. Ablation Study

We study the impact of data distributions in terms of imbalance (proportion of paired/partial instances for training) and the number of participating clients on the performance of the BlendFL framework. Specifically, the evaluated scenarios are described as follows:

- **Data distribution:** We evaluated five different ratios of paired-to-partial data distributions: (i) 90/10, (ii) 70/30, (iii) 50/50, (iv) 30/70, and (v) 10/90.
- **Number of clients:** We evaluated the scalability and robustness of BlendFL with a varying number of clients: 4, 8, 12.

Note that, for efficiency and interpretability, we conducted all these experiments with the S-MNIST dataset and compared BlendFL to the main FL paradigms, HFL and VFL, using implementations of FedAvg [6] and SplitNN [16], respectively.

V. RESULTS

This section reports the results for the tasks evaluated, including the real-world clinical tasks, generalization to additional dataset, and ablation studies.

A. Real-world medical tasks

Clinical conditions prediction. Table I reports the results for clinical conditions prediction on the clinical test set. Overall, BlendFL demonstrates superior performance for collaborative models, consistently outperforming all state-of-the-art FL

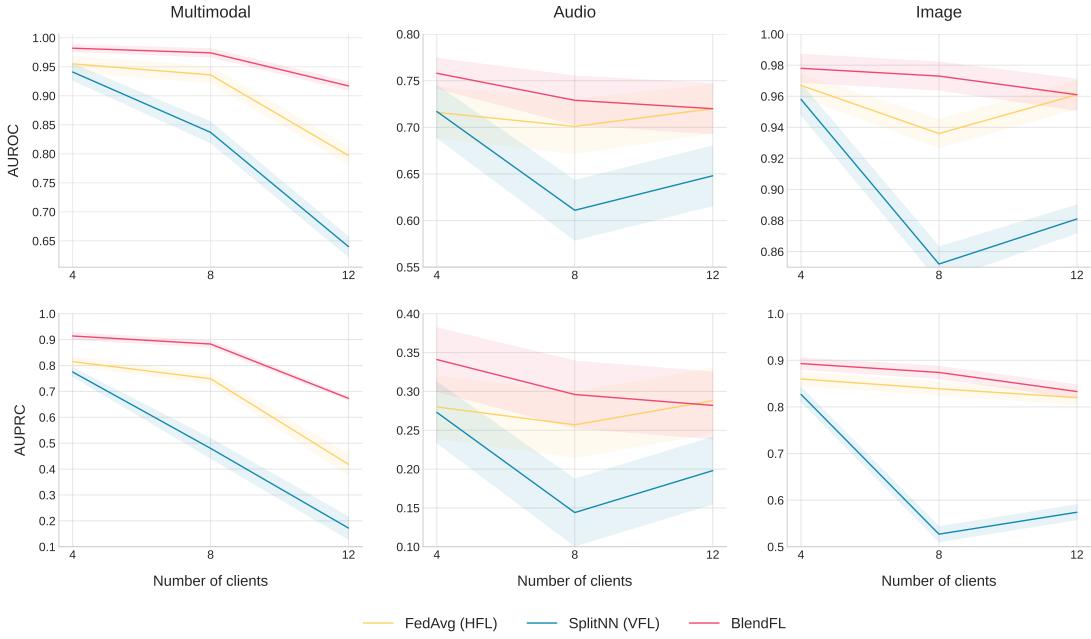


Fig. 4. **Number of clients.** Performance comparison of BlendFL and FL baselines for varying numbers of clients on the S-MNIST dataset.

baseline methods in terms of AUROC and AUPRC metrics. It also exhibits performance close to centralized learning across all metrics (upper bound for FL methods, with all data directly accessible for training). These results underscore the capability of BlendFL to handle complex, multilabel classification tasks effectively, approximating the ideal scenario of complete data availability.

In-hospital mortality prediction. Table II reports the results for in-hospital mortality prediction on the test set. As in the other task, BlendFL shows superior performance compared to all state-of-the-art FL baselines in terms of AUROC (unimodal and multimodal predictions), indicating its effectiveness in leveraging the full potential of the available heterogeneous data in a binary task. Regarding AUPRC, it outperforms all baselines for unimodal CXR and performs on par with other FL baselines for multimodal and unimodal EHR. Similarly, BlendFL achieves AUROC and AUPRC scores comparable to those of the centralized model.

B. Additional results on S-MNIST

Table III reports the performance results for BlendFL and baselines on the S-MNIST test set. As for the clinical tasks, BlendFL demonstrates superior performance for collaborative models, outperforming all state-of-the-art baseline FL methods across all metrics. Note that in this case, the performance gap between the baselines and BlendFL is greater than for the clinical tasks. It also achieves AUROC and AUPRC scores that are on par with those of centralized learning. These results demonstrate the efficacy of BlendFL in contexts where dataset size is constrained and modal variations are significant, highlighting its robustness and adaptability in various multimodal learning scenarios.

C. Model Convergence

Figure 3 reports the results for model updating strategies (BlendAvg and FedAvg) for varying intervals of local training epochs between updates. As the interval increases, the speedup on model convergence gained by using BlendAvg over FedAvg also increases, peaking at a 46% speedup when updates are made every 6 epochs of local training. This indicates that BlendAvg benefits from allowing local models to train more extensively before averaging, significantly reducing communication overhead while improving model convergence.

D. Ablation Studies

Data distribution. Figure 2 shows the impact of different data distributions in terms of paired/partial data on the performance of BlendFL and other baselines. Splits with a higher proportion of paired data favor VFL (SplitNN), reflecting their reliance on comprehensive feature sets per sample. Splits that favor partial data enhance HFL (FedAvg) performance, capitalizing on its strength in leveraging larger volumes of data for the same feature set. Additionally, BlendFL outperforms the baselines in each setting, effectively addressing the weaknesses of each one and maximizing model performance regardless of the distribution of the data.

Number of clients. Figure 4 reports the performance of BlendFL and the baselines when varying the number of clients. HFL methods generally perform better relative to VFL approaches as the number of clients increases, benefiting from the aggregation of more sample-diverse datasets. VFL approaches tend to underperform relative to HFL in scenarios with more clients due to the complexity of managing more extensive feature sets across common samples.

VI. DISCUSSION AND FUTURE WORK

Training collaborative models in real-world environments is a challenging task. While HFL and VFL enable training under certain conditions imposed on the clients, they fail to provide a framework for ill-defined scenarios where neither all features nor all samples are uniformly available across clients. To address this gap, we introduce BlendFL, the first FL framework that, unlike hybrid FL approaches, seamlessly integrates the strengths and full capabilities of both HFL and VFL for multimodal collaborative training. BlendFL enables collaborative training for diverse clients, allowing them to benefit from HFL, VFL, or both, regardless of their share of features and samples, and without restrictive requirements. Unlike VFL, it also enables local, independent inference, reducing dependency on the server for inference purposes and minimizing communication overhead.

We evaluate the performance of BlendFL using two datasets and three tasks. Our results demonstrate that BlendFL is superior to state-of-the-art FL frameworks under heterogeneous data conditions, consistently outperforming all state-of-the-art baselines across various datasets and tasks. In addition, BlendFL produces multimodal and unimodal encoders with performance on par with centralized models, considered the upper bound for FL. A key factor behind BlendFL's success is BlendAvg, a novel model parameter averaging strategy that shows faster convergence than FedAvg. The ablation studies, which evaluate BlendFL and baseline methods under varying data distributions (ratio of paired/partial data) and client numbers, highlight BlendFL's superior performance across various imbalance data and challenging conditions. Despite BlendFL's superior performance over all FL baselines, it is not immune to privacy threats. Future work should focus on integrating additional privacy measures into BlendFL, such as differential privacy, to strengthen data privacy and tighten security constraints within the framework.

VII. ACKNOWLEDGMENT

This work was supported by the NYUAD Center for Interacting Urban Networks (CITIES), funded by Tamkeen under the NYUAD Research Institute Award CG001, the Center for Cyber Security (CCS), funded by Tamkeen under NYUAD RRC Grant No. G1104, and the NYUAD Center for Artificial Intelligence and Robotics, funded by Tamkeen under the NYUAD Research Institute Award CG010. The research was carried out on the High Performance Computing resources at New York University Abu Dhabi.

This work was also supported by the Mubadala-NYUAD Collaborative Student Research Project Award.

REFERENCES

- [1] L. Yue, D. Tian, W. Chen, X. Han, and M. Yin, "Deep learning for heterogeneous medical data analysis," *World Wide Web*, vol. 23, pp. 2715–2737, 2020.
- [2] U. Milasheuski, L. Barbieri, B. C. Tedeschini, M. Nicoli, and S. Savazzi, "On the impact of data heterogeneity in federated learning environments with application to healthcare networks," in *2024 IEEE Conference on Artificial Intelligence (CAI)*. IEEE, 2024, pp. 1017–1023.
- [3] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare," *Nature medicine*, vol. 25, no. 1, pp. 24–29, 2019.
- [4] N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, S. Bakas *et al.*, "The future of digital health with federated learning," *NPJ digital medicine*, vol. 3, no. 1, pp. 1–7, 2020.
- [5] 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.
- [6] 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.
- [7] I. Dayan, H. R. Roth, A. Zhong, A. Harouni, A. Gentili *et al.*, "Federated learning for predicting clinical outcomes in patients with covid-19," *Nature medicine*, vol. 27, no. 10, pp. 1735–1743, 2021.
- [8] T. Chen, X. Jin, Y. Sun, and W. Yin, "Vaff: a method of vertical asynchronous federated learning," *arXiv:2007.06081*, 2020.
- [9] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. on Intel. Sys. and Tech.*, 2019.
- [10] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications," *Information processing & management*, no. 6, 2022.
- [11] 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*. PMLR, 2020.
- [12] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated learning with matched averaging," *arXiv:2002.06440*, 2020.
- [13] X. Zhang, W. Yin, M. Hong, and T. Chen, "Hybrid federated learning: Algorithms and implementation," *arXiv:2012.12420*, 2020.
- [14] Q. Li, C. Xie, X. Xu, X. Liu, C. Zhang *et al.*, "Effective and efficient federated tree learning on hybrid data," *arXiv:2310.11865*, 2023.
- [15] Y. Hu, D. Niu, J. Yang, and S. Zhou, "Fdml: A collaborative machine learning framework for distributed features," in *Proceedings of the 25th ACM SIGKDD Intl. Conf. on Knowl. Disc. & Data Mining*, 2019.
- [16] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," *arXiv:1812.00564*, 2018.
- [17] H. Gao, S. Ge, and T.-H. Chang, "Fedhd: Communication-efficient federated learning from hybrid data," *Journal of the Franklin Institute*, vol. 360, no. 12, pp. 8416–8454, 2023.
- [18] H. Zhang, J. Hong *et al.*, "A privacy-preserving hybrid federated learning framework for financial crime detection," *arXiv:2302.03654*, 2023.
- [19] D. Morales, I. Agudo, and J. Lopez, "Private set intersection: A systematic literature review," *Computer Science Review*, vol. 49, 2023.
- [20] 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.
- [21] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in neural information processing systems*, vol. 33, 2020.
- [22] J. Sun, Z. Xu, D. Yang, V. Nath, W. Li, C. Zhao, D. Xu, Y. Chen, and H. R. Roth, "Communication-efficient vertical federated learning with limited overlapping samples," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 5203–5212.
- [23] A. M. Elbir, S. Coleri, A. K. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, "A hybrid architecture for federated and centralized learning," *IEEE Trans. on Cogn. Comm. and Net.*, vol. 8, no. 3, 2022.
- [24] A. E. Johnson, L. Bulgarelli, L. Shen *et al.*, "Mimic-iv, a freely accessible electronic health record dataset," *Scientific data*, vol. 10, 2023.
- [25] A. E. Johnson, T. J. Pollard, S. J. Berkowitz, N. R. Greenbaum *et al.*, "Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports," *Scientific data*, vol. 6, p. 317, 2019.
- [26] N. Hayat, K. J. Geras, and F. E. Shamout, "Medfuse: Multi-modal fusion with clinical time-series data and chest x-ray images," in *Machine Learning for Healthcare Conference*. PMLR, 2022.
- [27] L. Khacef, L. Rodriguez, and B. Miramond, "Written and spoken digits database for multimodal learning," <https://zenodo.org/doi/10.5281/zenodo.3515934>, 2019.
- [28] Y. LeCun, "The mnist database of handwritten digits," <http://yann.lecun.com/exdb/mnist/>, 1998.
- [29] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv:1804.03209*, 2018.

- [30] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2016, pp. 770–778.