Knowledge and Information Systems
https://doi.org/10.1007/s10115-025-02356-y

REVIEW

# Vertical federated learning: a structured literature review

Afsana Khan[1] · Marijn ten Thij[1,2] · Anna Wilbik[1]

Received: 13 December 2023 / Revised: 28 November 2024 / Accepted: 23 January 2025

## Abstract

Federated learning (FL) has emerged as a promising distributed learning paradigm with an added advantage of data privacy. With the growing interest in collaboration among data owners, FL has gained significant attention from organizations. The idea of FL is to enable collaborating participants train machine learning (ML) models on decentralized data without breaching privacy. In simpler words, federated learning is the approach of "bringing the model to the data, instead of bringing the data to the model". Federated learning, when applied to data which is partitioned vertically across participants, is able to build a complete ML model by combining local models trained only using the data with distinct features at the local sites. This architecture of FL is referred to as vertical federated learning (VFL), which differs from the conventional FL on horizontally partitioned data. As VFL is different from conventional FL, it comes with its own issues and challenges. Motivated by the comparatively less explored side of FL, this paper provides a comprehensive overview of existing methods and developments in VFL, covering various aspects such as communication, learning, privacy, and applications. We conclude by identifying gaps in the current literature and proposing potential future directions for research in VFL.

**Keywords** Vertical federated learning · Privacy-preserving machine learning · Literature review

## 1 Introduction

The use of machine learning (ML) has enabled organizations to more quickly identify potentially profitable opportunities as well as risks that may be involved. As more and more data becomes accessible over time, there has been a corresponding rise in interest in the application

✉ Afsana Khan
a.khan@maastrichtuniversity.nl

Marijn ten Thij
m.c.tenthij@tilburguniversity.edu

Anna Wilbik
a.wilbik@maastrichtuniversity.nl

[1] Department of Advanced Computing Sciences, Maastricht University, Paul-Henri Spaaklaan 1, 6229EN Maastricht, The Netherlands

[2] Department of Cognitive Science and Artificial Intelligence, Tilburg University, Warandelaan 2, 5037AB Tilburg, The Netherlands

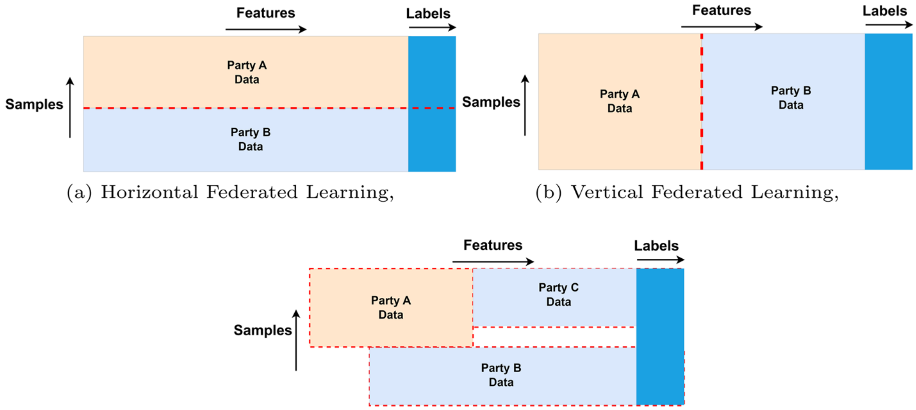
Published online: 13 February 2025

of machine learning across a variety of fields. For instance, in healthcare, the use of machine learning to analyze the health data generated by the growing number of wearable devices like smartwatches and fit bits is gaining momentum [11]. Moreover, the growing use of ML in financial systems has transformed industries and societies. From traditional hedge fund management firms to FinTech service providers, many financial firms are investing in data science and ML expertise [43]. ML has also made a significant contribution to the agriculture sector by creating new opportunities to unravel, quantify, and understand data-intensive processes in agricultural operational environments [79].

While organizations can benefit from applying machine learning techniques to their own data, doing the same with data from other comparable organizations could result in significant improvements to the existing organizational processes. In order to build sophisticated machine learning models for improving consumer service and acquisition, substantial emphasis has been placed on integrating data from various organizations, indicating the importance of collaboration. However, the traditional approach of bringing data located at different sites into a central server for training machine learning models is not always feasible as it can raise concerns. At present, sharing data among organizations has become critical due to concerns about privacy, maintaining competitive advantages, and/or other constraints. Data security and privacy are issues that are being prioritized not just by individuals or organizations but also by the larger society. The General Data Protection Regulations (GDPR), which the European Union put into place on May 25, 2018 [4] aims to protect users' personal privacy and data security. To address this issue, federated learning (FL), a new distributed learning paradigm, has recently received a lot of attention. FL allows collaboration among organizations to train machine learning models while ensuring that private data of these organizations are not disclosed [91]. Kairouz et al. [59] formally defined federated learning as

> "*A machine learning setting where multiple entities (clients) collaborate in solving a machine learning problem under the coordination of a central server or service provider. Each client's raw data is stored locally and not exchanged or transferred; instead, focused updates intended for immediate aggregation are used to achieve the learning objective*"

Depending on how the data is partitioned or distributed among organizations, FL can be classified into three scenarios; Horizontal, Vertical and Hybrid. Horizontal federated learning



(a) Horizontal Federated Learning,

(b) Vertical Federated Learning,

(c) Hybrid Federated Learning,

**Fig. 1** Types of federated learning

(HFL) is suitable in scenarios when organizations share the same attribute space but differ in samples (Fig. 1a). An example of HFL is a group of hospitals collaborating to build a ML model used to predict health risks for their patients, based on agreed-upon data. However, not all collaboration scenarios fit the HFL setting. Sometimes, in such collaborations, data can be partitioned vertically across different participants. VFL is suitable for scenarios where organizations have the same set of samples as their data but differ in feature space (Fig. 1b). It promotes collaboration among non-competing organizations with vertically partitioned data. In such cases, typically one organization has the ground truth, or labels, with some of the features of a number of samples. The rest of the participants take part in the federation by providing additional feature information of the same sample space but at the same time ensuring that their data is not disclosed directly to other participants. In return, these participants can be compensated with monetary and/or reputational rewards. An Example, where VFL is applicable, could be a telecom company collaborating with a home entertainment company (cable TV provider) or an airline collaborating with a car rental agency. Hybrid FL refers to the hybrid situation of horizontally and vertically partitioned data (Fig. 1c). In this scenario, the data owners hold different attributes for different data instances. However, hybrid FL has not yet been explored significantly in the literature.

VFL is a promising paradigm for privacy-preserving learning. Despite being a relatively new concept, recent contributions including peer-reviewed reviews Liu et al. [88] and preprints Li et al. [76], Yang et al. [148] have offered valuable insights into this field by offering overviews of VFL, covering its taxonomy, challenges, and applications (Table 1). Yet, there remains a need for a review that not only systematically evaluates the state-of-the-art and challenges in VFL but also follows a transparent and replicable process. Our literature review addresses this gap by employing a structured approach that provides a thorough and rigorous analysis as well as meticulously outlines each step of the review methodology, ensuring that our approach can be easily understood and replicated by researchers. A distinctive feature of our review is the introduction of a “Lifecycle Approach” to VFL which provides a comprehensive perspective, tracing VFL from its foundational concepts to practical applications. Furthermore, we identify additional gaps found in the literature compared to existing review papers and propose potential solutions to them which not only consolidates existing knowledge but also advances the current state of VFL and highlights key areas for future research & development. This review is not intended to provide an in-depth analysis of specific implementations of VFL. Instead, it aims to offer guidance to researchers exploring this field. For detailed information on VFL methods and techniques, readers should refer to the articles cited within.

## 2 Methodology

The review was planned, conducted, and reported in accordance with the SLR process proposed by Armitage and Keeble-Allen [6]. The SLR consists of five main steps including defining research questions, designing search strategy, selecting studies, extracting data and finally synthesis of data. Below we explain these steps in more detail.

### 2.1 Research questions

Taking into consideration the objective of the review, the following set of research questions were formulated as the initial stage of this SLR.

**Table 1** Comparison with similar reviews/surveys

| Aspect | This review | Liu et al. [88] | Yang et al. [148] |
|---|---|---|---|
| Methodology | Structured Literature Review (SLR), ensuring transparency and replicability | General overview, no clear replicable methodology | General overview, no clear replicable methodology |
| Organisation | Lifecycle Approach covering VFL from foundational concepts to real-world applications | Taxonomy categorizing VFL into key areas like communication efficiency, privacy, and data evaluation | Layered approach, dividing VFL components into layers: hardware, privacy-preserving primitives, secure algorithms, VFL systems, and applications |
| Coverage of VFL components | Covers all components of VFL (algorithms, communication, learning, privacy, valuation, and applications) uniformly and thoroughly discusses the most recent works | Covers most components but gives more details in areas like privacy, while providing less coverage on valuation and incentive mechanism, which is the management aspect of VFL | Discusses in detail areas like hardware and VFL algorithms, but does not mention learning challenges such as model fairness, limited training data, and feature selection |
| Open challenges & future directions | Model Drift, Fairness, Incentive Mechanism, Explainability, Dataset Availability | Interoperability, Trustworthy VFL, Automated and Blockchained VFL | Explains current developments, lacks clear future directions |

**Table 2** Search results

| Search terms | Google scholar | Web of science | IEEE Xplore | arXiv | No. of articles | Unique articles |
|---|---|---|---|---|---|---|
| "Vertical federated learning" | 158 | 53 | 66 | 61 | 338 | 226 |
| "Vertical" AND "Federated Learning" | 113 | 47 | 96 | 123 | 379 | 234 |
| "Vertical" AND "privacy-preserving federated learning" | 24 | 10 | 9 | 8 | 51 | 29 |
| "Vertical" AND "Heterogeneous Federated Learning" | 7 | 0 | 0 | 2 | 9 | 9 |
| Total no. of unique articles | | | | | | 271 |

- **RQ1:** What methods are currently employed in VFL, and how do they address its challenges?
- **RQ2:** What are the current applications of VFL?
- **RQ3:** What are the potential future directions for research in VFL?

## 2.2 Search strategy

After the formulation of the research questions, a plan was made to design the search strategy for the SLR. The search strategy includes the initial task of selecting literature databases. Kitchenham et al. listed several high-quality databases for searching research resources [67]. Since VFL is a trending research topic, there are many articles which are pre-prints. We chose to include both published (conference and journal papers) and pre-prints as a part of the SLR. The selected databases were Google Scholar, Web of Science (WoS), IEEE Xplore and arXiv. We experimented with different search terms on the chosen databases (Table 2). Finally, it was decided that the following search term would yield the most relevant results:

*("Vertical federated learning") OR ("Vertical" AND ("Federated Learning" OR "privacy-preserving federated learning" OR "Heterogeneous federated learning"))*

## 2.3 Study selection

The results obtained using the defined search strategy were filtered based on a set of criteria. Only the articles which met the following criteria were considered further for first-round screening:

- Published between 2016-2023
- Written in English language
- Availability of full text
- Title and abstract specifically mention the focus on vertical federated learning

After the initial screening, the chosen articles were checked for redundancy, which resulted in 174 unique articles. The unique articles were then investigated by going through the full text. Additionally, we used "snowballing" [117] technique to identify relevant articles. This was done by identifying relevant articles from the reference section of the previously selected articles. The use of the "snowballing" technique led to the finding of 9 additional relevant articles which were included in this review. These papers might have not appeared in the search due to database constraints, varied indexing terms, or inadequate keyword usage. The final articles which are included in this SLR were selected based on the fact that they answered the following questions

- Is the article relevant to VFL and not just general FL?
- Does the article provide an answer to any of the research questions?
- Does the article demonstrate a strong methodological approach either empirical or theoretical?
- Are the experiment setup and results properly documented?

In addition, survey papers were also considered where vertical federated learning had been addressed.

## 2.4 Data extraction

We extracted data from each of the included papers and organized it in a manner such that we could provide an analysis of the reviewed literature and use them for synthesis in the next section. The data which were extracted from the articles were title & year of publication, source of publication, research question/problem solved, proposed method, availability of theoretical analysis, dataset evaluated on and model evaluated with.

### 2.5 Data synthesis

As a last step of the literature review, we performed an analysis of 183 articles relevant to VFL and clustered them based on the problem they addressed and solved. A detailed review of these articles has been provided in the further section.

## 3 Results

We present the results of a structured literature review by reading and analyzing articles related to FL that are found in the four major databases mentioned earlier. These results of the study provide answers to the research questions that were presented in Section 2.1. To understand the research trend of VFL, we conducted statistics for the publication year of literature as shown in Fig. 2. Although the concept of federated learning was first introduced in 2016, research on FL until 2018 primarily concentrated on horizontal federated learning. But from 2019 a significant increase has been observed in the number of published articles focusing on VFL. Therefore, it can be concluded that VFL is still in its early stages of development.

In our analysis of the literature on VFL, we observed that over 60% of the papers reviewed were published in journals and conferences, suggesting that VFL has yielded mature and widely recognized results. This is indicative of the field's progression and acceptance in the academic and professional communities. Additionally, we noted that around 36% of the articles examined were pre-prints, primarily found on arXiv. This high proportion of pre-prints reflects the rapid evolution and ongoing development in the VFL domain, highlighting the cutting-edge nature of the research. While our review has considered pre-prints due to their relevance and the insights they offer into emerging trends, we have primarily focused on peer-reviewed papers. The inclusion of select pre-prints was based on their perceived quality and contribution to the field, ensuring that our review captures both the current state and the forward momentum of VFL research.

In our review, we systematically present the literature of VFL using a lifecycle approach (as displayed in Fig. 3), which details the progression from concept to application. The lifecycle begins with "VFL Foundations", where we explore the essential concepts and algorithms that form VFL. "VFL Development" takes us through how VFL models are improvised with aspects of communication, learning and privacy. "Evaluation and Management" involves

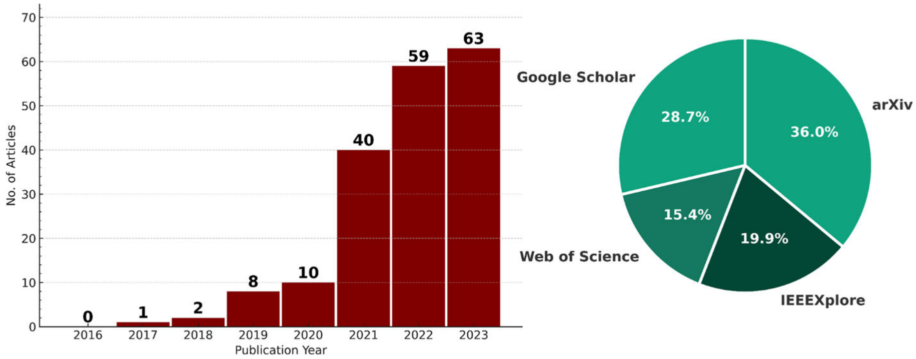


**Fig. 2** Overview of no. of articles found; yearly trend (left panel) & databases (right panel)

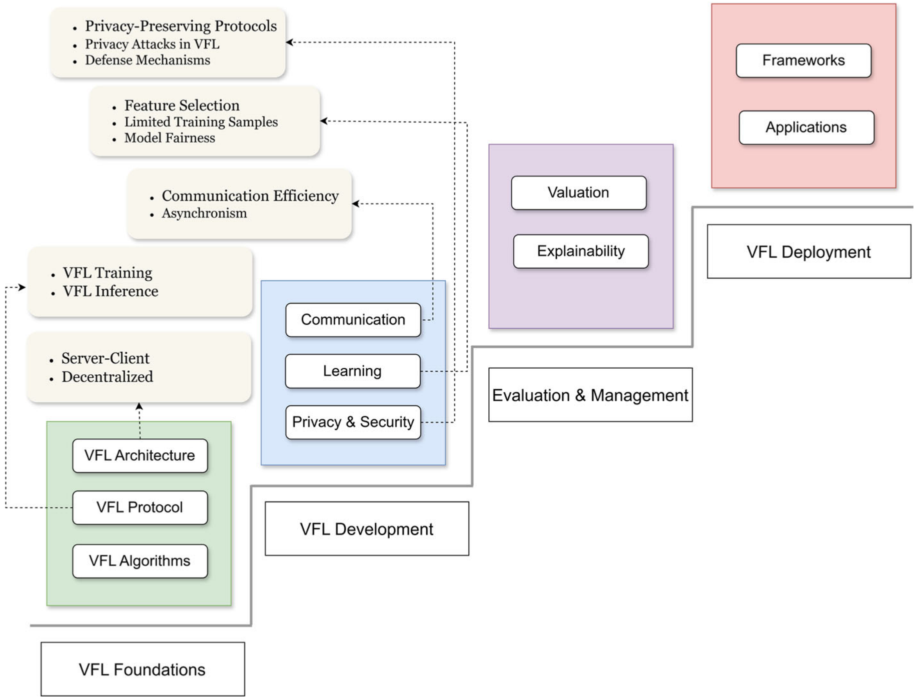


**Fig. 3** Literature of VFL as a lifecycle

steps such as measuring the contributions of participants, allocating incentives fairly, and ensuring the models and incentive mechanisms are understandable and transparent. Finally, “VFL Deployment” investigates the existing frameworks that facilitate VFL and the practical applications observed in the literature.

## 4 VFL foundations

Vertical federated learning is a distributed machine learning approach where a model is trained when the feature set of samples is distributed across multiple parties while also ensuring data privacy. This can be defined formally by considering a dataset $\{(x_i, y_i), i = 1, 2, .., n\}$ where $x_i \in \mathbb{R}^d$ represents the feature vector, $y_i$ denotes the output labels, and $d$ is the feature dimension. In VFL, this dataset is partitioned vertically across $M$ parties or clients, such that each party possesses a disjoint subset of the feature vector $x_{[i,m]}$, where $x_{[i,m]} \in \mathbb{R}^{d_m}$ and $d_m$ is the feature dimension for the $m$-th party. Furthermore, the model parameters for each party are represented as $\Theta = (\theta_1, \theta_2, .., \theta_M)$, where $\theta_m \in \mathbb{R}^{d_m}$. The goal of VFL is to minimize the following loss function:

$$\mathcal{L}(\Theta) = \frac{1}{n} \sum_{i=1}^{n} l \left( \sum_{m=1}^{M} x_{[i,m]} \theta_m, y_i \right) + \lambda R(\Theta)$$

where $l(.)$ and $R(.)$ represent the loss function and regularizer, respectively, and λ is a tuning parameter.

**Table 3** Server-client versus decentralized VFL

| Criteria | Server-client VFL | Decentralized VFL |
|---|---|---|
| Architecture | Central server | No central server |
| Communication | Through central server | Directly between parties |
| Single point of failure | Yes | No |
| Synchronization | Easier | More challenging |
| Security | Depends on server | Depends on protocol |
| Scalability | Can be limited | More scalable |
| Implementation | Easier | More challenging |

## 4.1 VFL architecture

In a VFL setting, two types of parties are involved [170]: the active/guest party and the passive/host party. The active/guest party holds both features and ground truth/labels, while the passive/host parties possess only features of the same instances. The main idea behind VFL is that the active party aims to collaborate with passive parties to leverage additional features in order to improve the model's performance, all while preserving the privacy of the data involved. This collaboration allows for a richer set of data to be used in training the model, potentially leading to more accurate and robust predictions.

Moreover, the architecture of VFL can be broadly categorized into server-client and decentralized. In a server-client VFL architecture, there is a trusted coordinator that facilitates the learning process, as proposed by Hardy et al. [46]. This coordinator is responsible for computing training loss and generating homomorphic encryption key pairs to ensure privacy. An example of a centralized VFL framework includes a setup with one trusted coordinator and two parties, where each party represents a single client. On the other hand, decentralized VFL architectures, as proposed by Yang et al. [149], He et al. [48], and Sun et al. [119], do not rely on a trusted coordinator. Instead, the active party takes the role of the co-ordinator and communication occurs between the active and passive parties, thereby reducing the system's complexity. This architecture has been further extended to include multiple collaborating parties or clients, as seen in works like Cheng et al. [23] and Zhao et al. [166]. In the literature, distinctions between server-client and decentralized VFL architectures have been outlined. However, several key differences have been identified through observations (Table 3).

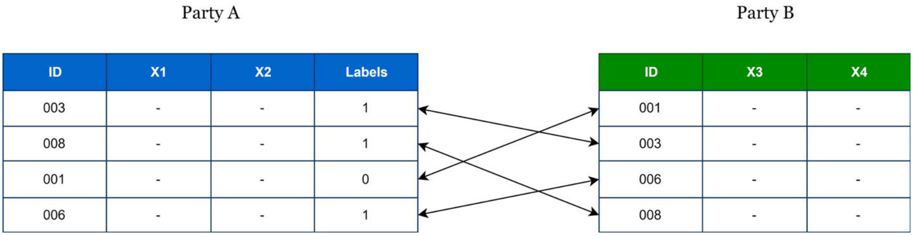
Each architecture has its own advantages and disadvantages, and the choice depends on the specific requirements of the federated learning scenario.

## 4.2 VFL protocol

In this section, we discuss the general VFL protocol which consists of two phases: training and inference.

### 4.2.1 VFL training

The training in VFL ensures that each participant's data remains private while still contributing to the improvement of the federated model. The key steps to the training process are explained as follows:



Party A

| ID | X1 | X2 | Labels |
|---|---|---|---|
| 003 | - | - | 1 |
| 008 | - | - | 1 |
| 001 | - | - | 0 |
| 006 | - | - | 1 |

Party B

| ID | X3 | X4 |
|---|---|---|
| 001 | - | - |
| 003 | - | - |
| 006 | - | - |
| 008 | - | - |

**Fig. 4** Data alignment between two parties in VFL

***Secure data alignment***

VFL requires the same set of data samples across different parties, and because these data samples may not be initially aligned, it is necessary to conduct an alignment process as the first step (Fig. 4). This step in VFL training involves determining overlapping samples among the participants. It ensures all parties work within the same sample space, maintaining consistency in the training. However, this identification of overlapping samples must be conducted in a privacy-preserving manner, safeguarding each participant's data.

A common technique to achieve this is private set intersection (PSI) [28, 37, 51] which is a privacy-preserving technique that is commonly used in vertical federated learning to enable parties to perform joint analysis on their data without revealing their private data. PSI allows two parties to compute the intersection of their private datasets without revealing the contents of the datasets. The basic idea behind PSI is that each party privately computes a hash function on their dataset and shares the hashes with the other party. The parties then compare their hashes to find the intersection of their datasets. The comparison is done without revealing any information about the actual data. After the intersection is computed, the parties can continue with the collaborative learning process using only the intersection of the datasets. Over the past few years, different hashing techniques for PSI have been proposed in Buddhavarapu et al. [14], Ion et al. [52], Chase and Miao [20], Lu and Ding [89] such as Bloom Filters and Oblivious Hashing. Bloom Filters are a probabilistic data structure that allow for efficient set membership testing, while Oblivious Hashing is a cryptographic technique that enables parties to privately compute hash functions without revealing the inputs to each other.

***Computation of intermediate results***

Following the secure data alignment, participants proceed to train on their aligned local data, typically utilizing gradient descent. During this phase, each party computes intermediate results before sending them to the central server (server-client architecture) or active party (decentralized architecture).

***Encryption of intermediate results***

In this step, encryption technique such as Homomorphic Encryption [153] is often employed to ensure that the intermediate results remain encrypted and secure during transmission and aggregation.

***Exchange of intermediate results and gradient computation***

The intermediate results are then securely transmitted to a central server or active party. Here, they are aggregated to compute gradients, which are then sent back to the parties. The parties use these gradients to update their local models through forward propagation. The exchange and update of gradients continue iteratively until the model converges to an optimal state, with each cycle improving the final model's accuracy and performance while preserving data privacy.

### 4.2.2 VFL inference

In VFL, the inference protocol differs significantly from HFL. In HFL, all parties possess the entire global model, while in VFL, they only hold parts of the federated model. The inference processes in VFL are collaborative, leveraging encrypted data exchanges to ensure privacy and security, unlike HFL, where inferences can be conducted independently at local sites. To evaluate and make predictions on new data in VFL, each party computes its segment of the model and communicates the results to a central server or an active party. The prediction phase is typically initiated by the active party, which sends out a prediction request with an input sample ID to the other parties. The passive parties provide their corresponding feature values for the given sample, and collectively, they execute a protocol to calculate the prediction outcome [90]. However, if a party cannot compute its part due to missing features for new data, it could lead to incomplete predictions or introduce inaccuracies. In this context, extending the Learning Using Privileged Information(LUPI) [102] paradigm in VFL could be an idea worth exploring when the test instances are systematically incomplete and the datasets are not shared.

## 4.3 VFL algorithms

Algorithm 1 demonstrates the training procedure for a general VFL model. A vertically federated logistic regression algorithm has been proposed in Zhu et al. [170], where participants train their local models using stochastic gradient descent (SGD) for computing intermediate results. Apart from logistic regression, VFL has also been proposed for other machine learning algorithms such as linear regression [41, 66], decision trees [65], random forests [84] and neural networks [167]. Liu et al. [84] proposed employing a combination of the CART tree algorithm, bagging techniques, and a third party to co-ordinator to design a vertical federated random forest algorithm. Wu et al. [137] developed a novel privacy-preserving solution for VFL, *Pivot* that enables secure decision tree training and prediction without relying on a trusted third party. They also address privacy leakages in plaintext decision tree models and extend their solution to ensemble models like random forest and GBDT. The framework *PyVertical* [108] makes the use of split neural networks for secure training of models with vertically partitioned data. It also incorporates the PSI protocol for secure alignment of samples before the training.

There has been some existing research conducted focusing on implementing the idea of VFL in extended scenarios. For instance, current VFL systems are developed on the assumption that the labels are possessed by only one client, i.e., the active party. However, there might be practical cases where multiple collaborating clients possess labels which arises the need to apply VFL in a modified manner. The Multi-VFL proposed by [97] makes use of split learning in a scenario where there are multiple data and label owners. Here, forward propagation is performed by the data owners on their corresponding partial models until the cut layer and then, their activations are sent to the label owners. These activations are concatenated by the label owners in order to complete their forward propagation. Subsequently, the losses are computed and back propagation is performed to compute the gradients. The gradients are then sent back to the data owners who are supposed to use them for completing their back propagation. Moreover, Zhu et al. [169] proposed a secure vertical FL framework PIVODL, to train gradient boosting decision trees (GBDTs) with data labels distributed on multiple devices. PIVODL presents a setting of training XGBoost decision tree models in VFL where each of the participating client holds parts of data labels that cannot be disclosed to others

**Algorithm 1** General VFL Training Algorithm

```
$M$ → Number of Clients
$T$ → Number of Communication Rounds
Training Data $X = \{X^1, X^2, \ldots, X^M\}$ partitioned by features across clients
Client 1 → Active Party/Guest Client
Client $2 \ldots M$ → Passive Party/Host Clients
Initialize local model parameters ${\theta_0}^m$ for each client $m \in \{1 \ldots M\}$
for $t = 1$ to $T$ do
    for each Client $m = 1$ to $M$ do
        Compute local intermediate results based on ${\theta_{t-1}}^m$ and local data $X^m$
        if $m \neq 1$ then
            Send local intermediate results from Client $m$ to Client 1
        end if
    end for
    Client 1 aggregates all received intermediate results to compute the gradient
    Compute the global gradient on Client 1
    Send the computed gradient to all Clients $2 \ldots M$
    for each Client $m = 1$ to $M$ do
        Update local models based on the computed gradient
    end for
end for
```

during the training process. A similar approach is also observed in Wang et al. [129] in order to deal with VFL when labels are distributed among multiple parties.

## 5 VFL development

This section of the VFL lifecycle addresses the development phase, concentrating on aspects such as enhancing communication efficiency, improving learning, and mitigating bias. It explores various strategies and methodologies that have been developed and implemented to optimize these components.

### 5.1 Communication

Efficient communication and asynchronous updates are critical in VFL to achieve efficient and scalable distributed machine learning. As VFL involves multiple parties collaborating to train a model while keeping their data private, effective communication protocols are necessary to ensure that parties can exchange information and update model parameters accurately and efficiently. Asynchronous updates enable parties to update model parameters independently, reducing the waiting time for other parties to complete their updates. This approach can improve the efficiency of the training process and allow parties to work independently, making VFL more scalable. This section discusses some of the existing approaches found in the studies to deal with communication efficiency and asynchronism in VFL settings.

#### 5.1.1 Communication efficiency

When following the conventional VFL approach, each of the passive/host clients share their updated gradients or intermediate results with the active/guest client during every training iteration. The total communication for each client can significantly increase over the course of hundreds or thousands of training iterations for very large data sets. As a result, the learning process might become inefficient due to communication cost and bandwidth constraints. Some existing researches [39, 87, 141, 145, 146] deals with the communication overhead problem in VFL by reducing the number of local model updates during training. A Federated Stochastic Block Coordinate Descent (FedBCD) algorithm was proposed by Liu et al. [87] for vertically partitioned data, wherein each party performs multiple local updates before each communication in order to reduce the number of client communication rounds significantly. Furthermore, Quasi-Newton method-based vertical federated learning systems [133, 146] proposed where descent steps scaled by approximate Hessian information are performed leading to faster convergence than Stochastic Gradient Descent (SGD)-based methods. This allowed a significant reduction in the number of communication rounds. Zhang et al. [157] proposed an algorithm to minimize the training time of VFL by adaptive selection of the number of local training updates for each party.

Another widely used strategy to achieve communication efficiency in VFL is the application of compression schemes to the data that is being shared among the clients. Castiglia et al. [18] demonstrate efficiency of VFL improves when the data to be transmitted such as gradients of clients are compressed (using quantization or sparsification) before sharing. Based on this idea, Yang et al. [147] proposed a method of gradient sharing and compression in VFL where only the gradients greater than a certain threshold are selected and then compressed by each of the clients before sharing in order to reduce communication bandwidth. Similarly, Li et al. [75] proposed an efficient vertical federated learning framework with gradient prediction and double-end sparse compression, where the compression occurs at the local models to reduce training time as well as transmission cost. Compression can also be directly applied on the local data of the host clients in a way that the compressed data contains relevant information of the local data. Later on these compressed local data are send to and aggregated by the guest client to train the model. Satisfactory outcomes to this approach have been observed in some of the research works where the compression of local data by extracting relevant information is done by using unsupervised techniques like Autoencoders [19, 63, 106], Feature Maps [111] and Representation Learning [139]. An overview of the approaches found in the studies for reducing communication overhead in VFL has been shown in Table 4.

### 5.1.2 Asynchronism

Vertical federated learning setting involves the collaboration of multiple clients having different features of a single data instance to train a machine learning model. But it is not always practical to assume that all the clients would be identical in terms of in storage, hardware, network connectivity, etc. Due to such variability among the clients there may be cases when one or more clients aren't participating in model updates at the same time typically resulting in asynchronous updates. Thus asynchronism can pose challenges in the proper functioning of VFL which is addressed by some of the research works by Zhang et al. [158], Gu et al. [44], Wei et al. [132]. A vertical asynchronous FL scheme was proposed [159] incorporating a backward updating mechanism and a bi-level asynchronous parallel architecture. The two-level parallel architecture: the inner level between active (available to share gradients) clients and the intralevel within each client. The updates at both levels are performed asynchronously which improves the efficiency and scalability. Moreover, Chen et al. [22] solved vertical FL in an asynchronous manner by allowing each client to run stochastic gradient algorithms without coordination of other clients. Thus, the temporary inactivity of any client does not pose any problem in the overall training.

Adaptive synchronization, developed for horizontal federated learning [85], could be adapted for vertical federated learning (VFL) to effectively manage asynchronism. This approach can

**Table 4** Approaches to reduce communication overhead in VFL

| | Article | Method | Model | Dataset |
|---|---|---|---|---|
| Modification in local updates | [87] | Stochastic Block Coordinate Descent with multiple update of local models | Logistic Regression, Neural Network | MIMIC-III, NUS-WIDE, MNIST, Default-Credit |
| | [146], [133] | Quasi-Newton Method | Logistic Regression | Default-Credit |
| | [145] | Eliminates need for peer to peer communication among clients by using functional encryption schemes | Linear regression, Logistic regression, linear SVM | Website phishing, Ionosphere, Landsat satellite, Optical recognition of handwritten digits, MNIST |
| | [141] | Allowed multiple local updates in each round by using alternating direction of multipliers | Convolutional Neural Network | MNIST, CIFAR-10, NUS-WIDE, ModelNet40 |
| | [39] | Cache enabled local updates at each client | Neural Network | $Criteo^5$, $zu^6$ |
| | [157] | Adaptive selection of local updates | Logistic Regression, Neural Network | a9a, MNIST, Citeseer |
| Compression | [18] | Arbitrary compression scheme on gradients of local models | Neural Network | MIMIC-III, CIFAR-10, ModelNet40 |
| | [147] | Transmission of selective gradients after compression | Logistic Regression | Default Credit |
| | [75] | Double-end sparse compression on local models | Logistic Regression, Neural Network | Default Credit, Insurance claim dataset |
| | [63] | Compression on local data using Autoencoders | Logistic Regression, SVM | Adult income, Wine-quality, Breast cancer, Rice MSC |
| | [106] | Compression on local data using Autoencoders | Logistic Regression | Bank loan dataset |
| | [19] | Compression on local data using Autoencoders | Neural Network | Adult income, Vestibular Schwannoma Dataset, The eICU Collaborative Research Database |
| | [111] | Compression on local data containing images using feature maps | Neural Network | CIFAR-10, CIFAR-100, CINIC-10 |
| | [139] | Compression on local data using unsupervised representation learning | Neural Network | NUS-WIDE, MNIST |

adjust update timings based on client activity, ensuring a more synchronized and efficient learning process. In addition, the domain of fault-tolerant systems designed specifically for VFL remains largely unexplored. Developing measures that can handle client failures or periods of inactivity, could significantly enhance the robustness and reliability of VFL.

## 5.2 Learning

Some learning approaches that has been used to address specific challenges in vertical federated learning have been discussed in this section.

### 5.2.1 Feature selection

To improve model performance and training time of machine learning models, feature selection is a widely used strategy. It refers to the practice of choosing a subset of relevant features (predictors and variables) for use in a model construction. Conventional feature selection methods like principal component analysis are simpler to apply in HFL setting compared to VFL. Since features are distributed across multiple clients, it becomes challenging to use the typical feature selection methods. Federated PCA has been proposed [25] where feature selection is achieved in a VFL setting at each client end by sharing of eigenvectors and eigenvalues of the host clients with the guest client. This was further extended in Cheung et al. [26] implementing vertically federated kernel PCA, VFedKPCA in order to capture nonlinear relationships among the features. A novel approach LESS-VFL [17] was proposed which utilizes group LASSO for VFL to efficiently eliminate irrelevant features in a way that minimizes the need for communication between parties. Furthermore, [35] proposed a VFL-based feature selection method *VFLFS* that leverages deep learning models as well as complementary information from features in the same samples at multiple parties without data disclosure. To improve this feature selection, non-overlapping samples have also been taken into consideration by utilizing autoencoders to learn their representations. Zhang et al. [160] introduce a Gini-impurity-based framework that enhances VFL with a privacy-preserving feature selection compatible with various model architectures. Building on this notion, [73] FedSDG-FS, refines the approach by incorporating a two-module strategy which begins with a Gini impurity-based feature importance, followed by an in-training selection of important features through a stochastic dual-gate mechanism. Zhang et al. [164] proposed a vertical federated feature selection framework PSO-EVFFS based on particle swarm optimization (PSO) with SecureBoost [23] for enhanced joint learning. A cooperative PSO algorithm has also been presented to fine-tune XGBoost hyperparameters and feature subsets simultaneously, supplemented by a feature importance-guided strategy to optimize PSO's initial search efficiency.

While some feature selection methods in VFL (Table 5) forgo privacy-preserving mechanisms under the assumption that not sharing raw data suffices for privacy, the absence of in-depth security analysis leaves potential vulnerabilities unaddressed. On the other hand, those employing Homomorphic Encryption for privacy face computational challenges. Future work should aim at exploring privacy solutions that minimize computational costs and evaluate the balance between maintaining privacy and ensuring the efficiency and effectiveness of feature selection method.

**Table 5** Summary of feature selection methods in VFL

| Feature selection method | Architecture | FS during training | Dependency on labels | Non-overlap utilization | Privacy protocol |
|---|---|---|---|---|---|
| FedPCA, VFedKPCA [25, 26] | Server-Client, Decentralized | No | Independent | No | No raw data shared |
| Federated LASSO Regularization [17] | Server-Client | Yes | Dependent | No | No raw data shared |
| VFLFS [35] | Decentralized | Yes | Partially Dependent | Yes | No raw data shared |
| FedSDG-FS [73] | Server-Client | Yes | Dependent | No | Partial HE |
| MMVFL [36] | Server-Client | Yes | Partially Dependent | No | No raw data shared |
| Gini-impurity FS [160] | Server-Client | No | Dependent | No | Secret Sharing, HE |
| PSO-EVFFS [164] | Decentralized | Yes | Dependent | No | HE |

### 5.2.2 Limited training samples

It is more practical to apply any of the PSI protocols before the implementation of VFL in a real-world application in order to determine the common set of data records. But a crucial fact that is to be considered is that there might not always be enough common or overlapping samples of data available among the clients. VFL might not produce satisfactory results due to lack of sufficient data. To address this problem a solution could be to expand the training data which has to be done also in a privacy preserving manner. A data augmentation method, FedDA proposed by [156] uses generative adversarial network (GAN) to generate more overlap data by learning the features of finite overlap data and many locally existing non-overlap data among the clients. Similarly, the semi-supervised learning approach FedCVT in Kang et al. [61] improves the performance of the VFL model with limited aligned samples by expanding the training data through estimation of representations for missing features and predicting pseudo-labels. Some other approaches to tackle the issue with limited samples include determining inferences from non-overlapping data by using Federated Transfer Learning [40] and Oblivious Transfer [107]. While utilizing GANs and supervised or semi-supervised learning methods for data generation and augmentation in VFL is a promising approach, it opens up a research direction in ensuring that the generated data is of high quality, truly representative, and free from bias. Future research could focus on developing methods and metrics to evaluate the quality and bias in generated or augmented data to improve the robustness and fairness of VFL.

### 5.2.3 Model fairness

Machine learning models in some cases may manifest unexpected and erratic behaviors. These behaviors when have undesirable effects on users, the model can be deemed as "unfair" based on some criteria. The existing bias in the training data is one of the key causes of a model becoming unfair. As real-world data encodes bias on sensitive features such as age, gender, and so on, VFL models may adopt bias from data and become unfair to particular user groups [136]. Again due to features being decentralized across different parties, applying existing fair ML methods to VFL models becomes challenging. Qi et al. [104] have addressed this issue by proposing a FairVFL framework where unified and fair representations of samples are learned based on the decentralized features in a privacy-preserving way. In order to obtain fair representations, adversarial learning has been used to eliminate bias from the data. A superior performance in training fair VFL model was achieved [83] in which the fair learning task was modeled as a non-convex constrained optimization problem. The equivalent dual form of the optimization problem was considered, and subsequently, an asynchronous gradient coordinate descent ascent algorithm was proposed to solve the dual problem.

In VFL, a persistent challenge lies in accurately identifying sensitive features during the training phase due to the decentralized and private nature of the data. Furthermore, determining what constitutes 'fairness" and establishing universally accepted criteria or metrics for fairness assessment remains a complex issue. Future directions for enhancing fairness in VFL should thus prioritize optimizing the identification of sensitive data. Additionally, there is a pressing need to establish benchmarks for fairness, which could serve as standardized reference points to guide and evaluate efforts in mitigating bias in VFL models.

## 5.3 Privacy and security

Federated learning ensures privacy of data while federation among clients since training of models occurs locally and data never leaves their local sites. In vertical federated learning process, the federated model is trained by sharing of gradients or intermediate results among clients. However, some studies conclude that, there are still possibilities of sensitive private data being leaked through the local gradients in Aono et al. [5], and participants' data can be inferred through a generative adversarial network during the prediction stage in VFL [90].

### 5.3.1 Privacy-preserving protocols

According to recent studies on VFL, the widely used privacy-preserving protocols include Homomorphic Encryption (HE), Secure Multi-Party Computation (SMPC) and Differential Privacy (DP).

***Homomorphic encryption***
Homomorphic encryption (HE) [153] is a cryptography technique which allows specific types of computations to be carried out on ciphertexts and generates an encrypted result which, when decrypted, matches the result of operations performed on the plaintexts. For the purpose of encrypting intermediate results (e.g., gradients), VFL typically utilizes additively homomorphic encryption like Paillier [101]. Additively homomorphic encryption allows participants to encrypt their data with a known public key and perform computation with the encrypted data by other participants with the same public key. The encrypted data need to be sent to the private key holder so that it can be decrypted. A secure cooperative learning framework was proposed [46] for vertically partitioned data using additional HE. The framework was evaluated to be precise as the non-private solution of centralized data. Moreover, it scaled to problems with large number of samples and features. Similarly, [30] proposed a privacy-preserving DNN model training scheme based on homomorphic encryption is for vertically segmented datasets. Moreover, several other studies [99, 146, 149] also have used HE as a privacy preserving protocol while proposing vertical federated learning approaches.

***Secure multi-party computation***
Secure multi-party computation allows a set of parties to jointly compute a function over their inputs while keeping those inputs private. SMPC is particularly useful in scenarios where the raw data cannot be shared due to privacy concerns, and only the computed results, such as model updates or predictions, need to be shared which makes it ideal for vertically federated settings. Nguyen Duy et al. [98] applied SMPC to VFL by employing a 3-party secure computation protocol (3PC) allowing private data computations to build Symbolic Regression models through secret sharing. A similar approach was adopted by Shi et al. [112], where an efficient multi-participant VFL method was proposed. Utilizing SMPC instead of HE turned out to be comparatively computationally efficient.

***Differential privacy***
Differential privacy is a privacy-preserving protocol for bounding and quantifying the privacy leakage of sensitive data when performing learning tasks. It relies on adding noise to original data or training results to protect privacy. Too much noise can degrade the model's performance, while too less data can breach privacy. Hence, a balance between performance and privacy has to be achieved here. Wang et al. [126] designed a DP-based privacy-preserving algorithm to ensure the data confidentiality of VFL participants. The algorithm, when implemented, was quantitatively and qualitatively similar to generalized linear models, learned in

an idealized non-private VFL setting. A multiparty learning framework for vertically partitioned datasets proposed in Xu et al. [144] achieves differential privacy of the released model by incorporating noise to the objective function. In this case, the framework requires only a single round of noise addition and secure aggregation. In addition to using DP during model training, it can also be used during the model evaluation phase in VFL since there is also possibility of leaking private label information. Sun et al. proposed two evaluation algorithms in Sun et al. [122] that accurately compute the widely used AUC (area under curve) metric when using label DP [42] in VFL.

Homomorphic Encryption and Secure Multi-party Computation offer strong privacy in data analysis but come with a significant computation. Optimizing these methods is essential, involving algorithmic improvements and efficient implementation. Differential Privacy, on the other hand, is more computationally efficient but faces a trade-off between privacy and data utility, as the noise added for privacy can reduce data accuracy.

### 5.3.2 Privacy attacks in VFL

Attacks in federated learning refer to malicious attempts by an attacker to manipulate or compromise the integrity of the federated learning process. Label inference attacks and feature inference attacks are the most commonly explored adversarial attacks in VFL. A label inference attack in vertical federated learning is a type of privacy attack where a malicious participant tries to infer the labels (i.e., the output values) associated with the training data of other participants, with the aim of obtaining sensitive information. Because no raw data is shared between the two parties, VFL initially appears to be private. At the end of a passive party, a considerable amount of information still exists in the cut layer embedding that can be used by the active party to leak raw data [38]. Fu et al in Fu et al. [38] explored the possibilities of inferring labels exploiting the gradient update mechanism in VFL and as well as inferring labels from the bottom model trained locally by each participant to embed the input features to a latent space, which avoids the raw features being directly sent to the server. [173] also showed that private labels could be reconstructed with high accuracy by training a gradient inversion model even with HE-communication. Moreover, [121] proposed a label inference method where it was possible to steal private labels effectively from the shared intermediate embedding even though label differential privacy and gradients perturbation were applied.

On the other hand, a feature inference attack is a type of privacy attack that aims to infer private information about a party's data based on the model's output. In vertical federated learning, an attacker may try to infer a party's private features by analyzing the model's output on the shared features. For example, an attacker may try to infer a person's medical condition based on their zip code and medical procedures, which are shared between different parties. [151] proved theoretically and experimentally that, it is possible to perform a reconstruction attack on the features of the parties by applying a robust search-based attack algorithm unless the feature dimension is considerably large enough. Furthermore, [90] proposed a general feature inference attack that learns the correlations between the features of the adversary and the attack target's features based on multiple predictions accumulated by the adversary.

### 5.3.3 Defense mechanisms

Defense mechanisms are crucial for vertical federated learning because this approach involves training machine learning models on data that is distributed across different entities or organizations. This means that the data is often sensitive and private, and its exposure could lead to significant risks, such as identity theft, financial fraud, or discrimination. Mainstream defense mechanisms like adding noise to gradients [171], gradient compression [82], and randomly pruning part of the gradients to zero [113] are not able to mitigate possible label information leakage in VFL settings [38]. [121] proposed adding an additional optimization goal at the label party and limiting the label stealing ability of the adversary by minimizing the distance correlation between the intermediate embedding and corresponding private labels. Similarly, a dispersed training framework was also been proposed in Wang et al. [131] where secret sharing had been used in order to break the correlation between the bottom model and the training data. In addition, [131] also described a customized model aggregation method such that the shared model can be privately combined, and as well as the training accuracy in ensured by the linearity of secret sharing schemes. Furthermore, [173] demonstrated that using confusional autoencoder (CAE); a technique based on autoencoder and entropy regularization and its variant DiscreteSGD-enhanced CAE could successfully block label inference attacks without significantly hampering the accuracy of the main task.

Additionally, [105] proposed HashVFL which is a hashing-based VFL framework to deal with feature reconstruction attacks in a VFL setting. The one way nature of hashing enables blocking of all attempts to recover data from hash codes. Efficiency of HashVFL in mitigating reconstruction attacks have also been demonstrated through experimental results. An adversarial training- based framework for VFL has been designed in Sun et al. [120] which simulates the game between an attacker (i.e., the active party) who actively reconstructs raw input from the cut layer embedding and a defender (i.e., the passive party) who aims to prevent the input leakage. A similar approach is observed in Luo et al. [90] which deals with malicious attack by active party during model evaluation stage of VFL.

## 6 Evaluation and management

This section discusses the literature focusing on the evaluation and management aspects after the successful implementation of VFL, such as the valuation of data from the parties, allocation of incentives, and the explainability of the federated model.

### 6.1 Valuation

The concept of fairness in FL also includes collaboration fairness which implies the incentive mechanism in a FL setting. Since FL is based on collaboration, rewarding mechanism is crucial for the allocating rewards to current and potential participants of FL. To achieve that, FL needs a fair evaluation mechanism to give agents reasonable rewards. Moreover, to analyze business value of VFL in real-world application determining a proper incentive mechanism for the participating clients is crucial since not doing do may result in lack of motivation from the clients to collaborate. The Shapley value (SV) [109] is a provably fair contribution valuation metric originated from cooperative game theory. Song et al. [115] introduced computing SV in HFL setting in order to evaluate the contribution of participants. The conventional approach for computing the SV in federated learning involves retraining and evaluating models on all possible coalitions of participants. This process is straightforward in HFL because the global model is formed by aggregating the local models from each participating client. The aggregation process is additive, meaning that each client's contribution can be seamlessly added to or removed from the global model. Therefore, by simply aggregating local models from various subsets of participants, the SV can be computed effectively for each participant. On the other hand, in VFL, clients hold different features for the same set of samples, and the global model is formed by concatenating these feature-based local models rather than aggregating them. Each client's model contributes a different aspect of the feature space, and the combination of these models is not straightforwardly additive. Instead, the global model relies on a composite structure that integrates these diverse feature sets. This global model is never shared with a server or co-ordinator. Thus, calculating SV by direct concatenation of local models from only a subset of clients is not applicable here. Since the conventional approach of computing SV suffers from the exponential time required for retraining models with all possible coalitions of participants, many studies [115, 127, 130] solved this problem by introducing approximation methods. Song et al. [115] proposed two gradient-based methods to reconstruct models during federated learning in order to avoid extra training. The first updates the initial global model with gradients from different rounds and assesses contributions based on the performance of these reconstructed models. The second updates the global model from the previous round with current round gradients and then aggregates the contribution indexes from multiple rounds using weighted sums. However, this approach requires testing model performance exponentially, leading to high computational costs. Wang et al. [130] measured the contribution of participants by calculating federated Shapley value, *FedSV* for each participant. This approach eliminated the need to retrain models; rather, the Shapley values were computed from local updates during each iteration and then aggregated after completing all iterations to obtain the final federated Shapley value. Again, this approach is not applicable to VFL due to the non-additive nature of local models.

Some works [34, 127] have tailored different techniques for data valuation in VFL settings. Wang et al. [127] proposed using situational importance (SI) to compute the difference between the actual contributions of feature values and their expected contributions in a model. This approach, particularly suited for additive models like linear regression, leverages the direct influence of individual features on model predictions. However, in case of VFL, where data is siloed and sharing across parties is restricted to preserve privacy, knowledge of these expected values for computing SI is not feasible. Moreover, this approach is only applicable for synchronous settings. Fan et al. [34] further extended this idea to both synchronous and asynchronous vertical federated algorithms. Wang et al. [128] proposed estimating SV for participants based on locally computed results that are securely aggregated without revealing individual data entries. This process has been enhanced by a re-weighting mechanism that dynamically adjusts participant weights based on their contributions to improve the model's convergence speed and accuracy. However, for VFL, homomorphic encryption has been used as a privacy protocol which can lead to high computational overhead in case of large number of participants. Zhang et al. [165] extended the idea of contribution measurement of participants in VFL by incorporating the data pricing model based on Stackelberg with the hosts as the leader and the guest as the follower. This approach provided managerial guidance on revenue distribution for FL platform owners based on their contributions.

From the literature, it is evident that while the SV is extensively used for contribution measurement and data valuation in FL, most works are not specifically tailored for application in VFL or lack sufficient experimental validation in VFL settings. Alternatively, exploring other concepts from cooperative game theory might offer viable solutions for contribution measurement and incentive mechanisms in VFL, potentially reducing computational complexity while maintaining fairness in reward distribution.

### 6.2 Explainability

An area of study that has gained a great deal of attention recently is explainable artificial intelligence (XAI). Models must be explainable in high-stakes applications like healthcare or finance when there is a strong need to justify decisions made. The same applies in case of VFL as well. However, when exploring through the literature, only a handful of research was found focusing on the explainability aspect of VFL. Kang et al. [60] proposed a method to enhance VFL prediction models by grouping original features into explainable groups and learning meaningful high-order features for improved interpretability and transferability. Moreover, Chen et al. [21] proposed a credible federated counterfactual explanation (CFCE) approach, utilizing a federated counterfactual loss to analyze the importance of features distributed across multiple parties in VFL models. However, proper privacy analysis for these methods would be required. Future research could focus on balancing the need for clear model interpretation with strict privacy requirements. Developing methods that offer meaningful insights without compromising the privacy of the distributed data involved in VFL poses a significant challenge and is a crucial area for exploration.

## 7 VFL deployment

Google initiated project in 2016 to establish federated learning among Android mobile users [13]. The goal was to improve the keyboard input prediction quality, while at the same time ensure the security and privacy of users. In the later stage, many use cases were addressed in several surveys and studies where FL could be implemented.

### 7.1 Frameworks

There have been significant efforts in designing federated learning frameworks for industrial usage but most of them are still in their development stage. A few of the existing FL frameworks support VFL either completely or partially.

**FATE** [86], developed by WeBank and released in 2019, is a business-ready FL framework for both horizontal and vertical settings. It integrates privacy-preserving technologies such as Homomorphic Encryption and Secure Multi-Party Computation. Despite its robust capabilities, including a variety of pre-built modules for different machine learning algorithms like gradient-boosted decision trees and tools such as FATEBoard for live visualization and FATEServing for model deployment, FATE's complexity and limited flexibility in terms of customization pose challenges. It supports major deep learning libraries like PyTorch and TensorFlow, enhancing its applicability in numerous applications.

**FederatedScope** [142] developed by Alibaba Group employs an event-driven architecture that enhances flexibility in the development and execution of horizontal and vertical FL processes, accommodating both synchronous and asynchronous training strategies. It supports multiple machine learning libraries such as PyTorch and TensorFlow via sophisticated message translation mechanisms. It also integrates privacy-enhancing technologies including Differential Privacy, Secure Multi-Party Computation, and Homomorphic Encryption. HE is implemented using the Paillier scheme specifically for linear models within vertical FL settings, enhancing its capability to handle sensitive data securely.

**NVFLARE** [110] is an open-source, business-ready FL framework developed by Nvidia, specifically designed with healthcare applications in mind for both horizontal and vertical settings. Released in 2021, it supports a wide range of machine learning models, including neural networks and tree-based models. It features robust project management capabilities, supports parallel job execution, and offers advanced privacy features such as differential privacy and homomorphic encryption. While it provides substantial flexibility and comprehensive functionalities, getting used to its architecture may require some effort. The framework includes orchestration through Docker-compose and a user-friendly dashboard for effective monitoring and management of FL projects.

**Flower** [10] is a framework-agnostic FL platform that supports both horizontal and vertical federated learning, developed by the German company Adap. It facilitates easy migration of a wide range of ML models, including neural networks, into the federated setting. It offers robust documentation, example projects, and experimental support for Differential Privacy. Despite its flexibility and scalability, Flower's current version lacks advanced privacy features and deployment functionalities.

**FedML** [47] is a flexible, distributed FL framework designed to facilitate the easy implementation of new algorithms across various network topologies and real-world hardware platforms. It separates distributed communication and model training into distinct modules, with the former based on MPI and supporting advanced privacy protocols and the latter built on PyTorch. FedML is notable for its client-oriented programming interface that promotes flexible distributed computing, crucial for scenarios constrained by GPU memory and training times. Additionally, it supports practical FL applications on devices like smartphones and Raspberry Pis, emphasizing real-world usability and scalability.

**PySyft** [172] developed by OpenMined primarily supports PyTorch and TensorFlow, and ensures privacy through Differential Privacy and Secure Multi-Party Computation. PySyft's integration with PyGrid facilitates network management across operating systems using containers. However, its support for vertical federated learning requires the use of extra library PyVertical [108], introducing dual dependencies that may complicate implementation. Despite its variety of features, documentation inconsistencies pose challenges, although ongoing development and a proactive community signal future enhancements.

**PaddleFL** [8] is designed to facilitate the easy replication and comparison of FL algorithms across various applications like computer vision and NLP. It supports both horizontal and vertical federated learning, integrating advanced strategies such as Multi-task and Transfer Learning. PaddleFL performs well in large-scale deployments with its efficient use of Kubernetes for job scheduling. Additionally, it offers Secure Multi-party Computation for privacy-preserving training and predictions, making it suitable for secure data sharing across different organizational setups.

**FedTree** [77] is designed for tree-based models like GBDT, adapted for both horizontal and vertical federated learning environments. It employs a unique histogram-based sharing approach. Key features include secure multi-party computation protocols and privacy mechanisms like homomorphic encryption and differential privacy to protect data during training. FedTree is structured to optimize communication and computation processes, supporting effective federated learning deployment across diverse data environments.

**CrypTen** [68], developed by Facebook AI research, is a privacy-preserving machine learning framework built using PyTorch, enabling researchers and developers to train models on encrypted data. It primarily utilizes Secure Multi-Party Computation for encryption, integrating closely with PyTorch to streamline the transition from traditional to secure environments. This approach not only enhances usability but also ensures that computations remain efficient and compatible with modern machine learning operations and GPU support.

As observed from Table 6, most FL frameworks adhere to a server-client architecture, which may limit scalability and flexibility compared to FedML and FederatedScope, which also support decentralized architectures. In terms of model support, frameworks like FedTree focus on specific models such as tree-based models, offering specialized tools but lacking the versatility seen in FederatedScope, Flower or Crypten that support a wider range of models including GNNs and neural networks. Handling asynchronous updates, a critical feature for large-scale deployments, are notably absent in most of the frameworks except NVFLARE and FederatedScope, potentially reducing their effectiveness in environments with highly variable network conditions. Privacy protocols vary widely: FederatedScope offers comprehensive privacy solutions including HE, SMPC and DP, providing robust data protection. However, the inclusion of multiple complex protocols can increase computational overhead and complexity. On the other hand, frameworks like Flower and Crypten may offer fewer privacy options, which might simplify operations but at the cost of reduced data security. Therefore, the choice of an FL framework greatly depends on specific use cases and requirements. Each framework has its own unique strengths and weaknesses, making them suitable for different scenarios. For instance, NVFLARE focusing on healthcare with its emphasis on DP may be ideal for sensitive medical data, while the flexibility of FedML and Flower might be better suited for academic research environments exploring novel federated learning algorithms. Selecting the right framework requires careful consideration of the intended application, required model complexity, necessary privacy safeguards, and network architecture.

## 7.2 Applications

Federated learning is a cutting-edge modeling mechanism which is capable of training a unified machine learning model using decentralized data while maintaining privacy and security. Thus, it has a promising application in financial, healthcare and many other industries where direct aggregation of data for training models is not feasible due to concerns like data security, privacy protection and intellectual property. Most applications are focused on horizontal federated learning and it is quite difficult to observe VFL being used in real-world applications yet. However, in our literature review, we found a limited number of studies proposing and as well as implementing VFL in applications.

***Healthcare***

Sun et al. [123] considered a scenario involving two hospitals; Inner and Outer hospital possessing records of daily performance and clinical test of patients. Due to patients privacy regulations, it was not allowed to share raw data among the hospitals even though they had records of the same patients. Hence, the Inner- and Outer-hospital information had been bridged via vertical Federated Learning for perioperative complications prognostic prediction. Cha et al. [19] proposed using a communication-efficient VFL on the Schwannoma dataset for predicting hearing impairment after surgery by representing the sensitive medical data into compressed latent representations for privacy. Tang et al. [124] combined the techniques homomorphic encryption and secret sharing to implement intention-hiding vertical federated learning (IHVFL) which predicted breast cancers efficiently and with better accuracy. Islam et al. [54] applied VFL to four healthcare applications; predicting in-hospital mortality, forecasting duration of patient stay, classifying phenotype and, detecting decom-

**Table 6** Comparison of federated learning frameworks supporting VFL

| Framework | Communication architecture | | Asynchronous update | Models supported | Privacy & Security protocols | | |
|---|---|---|---|---|---|---|---|
| | Server-client | Decentralized | | | HE | SMPC | DP |
| PySyft | ✓ | × | × | Regression, NN | × | × | ✓ |
| FATE | ✓ | × | × | Regression, Tree-based | ✓ | ✓ | × |
| PaddleFL | ✓ | × | × | Regression, NN, SVM | × | ✓ | ✓ |
| FedML | ✓ | ✓ | × | Regression, NN | × | ✓ | ✓ |
| NVFLARE | ✓ | × | ✓ | NN, Tree-based | × | × | ✓ |
| FederatedScope | ✓ | ✓ | ✓ | Regression, NN, GNN | ✓ | ✓ | ✓ |
| Flower | ✓ | × | × | General ML Models, NN | × | × | ✓ |
| Crypten | ✓ | × | × | General ML Models, NN | × | × | ✓ |
| FedTree | ✓ | × | × | Tree-based (GBDT) | × | × | ✓ |

pensation using the MIMIC-III dataset. Zhang et al. [160] used VFL with a secure feature selection technique to reduce communication overheads for training on real-world medical datasets. Some works [118, 135] have used privacy-preserving protocols for analysis of healthcare data in a vertically partitioned scenario and mentioned the future implementation of VFL in these use cases. Worm et al. [135] implemented the secure multi-party computation protocol to analyze cancer patient data in a vertically partitioned setting to predict the chances of survival of patients after diagnosis.

***Finance***

Vertical federated learning when applied to financial institutions is also able to boost profits by collaborating data among them maintaining privacy [33, 81]. Liang et al. [81] implemented a Hetero Secure Boost Tree (HSBT) algorithm for training VFL model which enabled collaboration between telecom and finance companies improving the success rate of marketing of a commercial bank. Efe [33] proposed a Multi-Institutional Credit Scoring (MICS) for VFL to enable cooperation among different telecom and banks for better credit scoring of consumers in a privacy-preserving manner. Cheng et al. [24] discussed a use case *FedRiskCtrl* which involved designing a risk control model for small and micro-enterprise loans where invoice agencies collaborated with a Webank in a VFL setting. Li et al. [74] proposed a privacy-preserving sampling strategy for VFL in case of imbalanced financial data by having participants provide partial neighbor information for each sample during the intersection stage and, filtering out controversial negative samples. Abadi et al. [1] discussed using VFL financial fraud detection through the collaboration of third party financial service providers and banks.

***E-commerce***

VFL has also been useful in the field of e-commerce as observed in Zhang and Jiang [155] where a method based on clustering and latent factor model under the vertical federated recommendation system was implemented. Taking into account the diversity of a large number of different users in each participant and the complexity of the matrix factorization of the user-item matrix, the users were clustered to reduce the dimension of the matrix and improve the accuracy of user recommendations. Similarly, efficient online advertising was achieved through the application of VFL [78]. Mai and Pang [92] developed a vertically federated graph neural network (VFGNN)-based recommender system that has competitive prediction accuracy with existing privacy-preserving GNN frameworks along with privacy protection for user interaction information. Cao et al. [16] studied a contextual bandit learning problem [2] for recommendation in the vertical federated setting. In this approach, an encryption scheme named orthogonal matrix-based mask mechanism was applied to bandit algorithms for privacy-preserving online recommendation. proposed a differential privacy (DP)-based federated cross-domain social recommendation (FCSR) algorithm to protect user connections in a cross-domain social recommendation system. It aims to facilitate collaboration between information-oriented and social-oriented websites by securely integrating different types of user data.

***Other applications***

In [168] a VFL scheme has been developed for the purpose of human activity recognition (HAR) across a variety of different devices from multiple individual users by integrating shareable features from heterogeneous data across different devices into a full feature space. Liu et al. [84] proposed an Aligning, Integrating and Mapping Network (aimNet) which enhances image representations by combining data from various vision-and-language tasks. It efficiently used data across different tasks through VFL for more powerful image representations which allows for data privacy preservation. Zhang et al. [163] introduced a vertical federated learning-based cooperative sensing (VFL-CS) scheme for cognitive radio networks to avoid malicious exploitation of sensitive sensing results by keeping the data at the local points. A special use case of VFL was observed in the aviation sector [45] where a flight delay prediction model based on federated learning was designed by integrating horizontal and vertical federated frameworks.

While VFL has shown promise in sectors such as healthcare, finance, and e-commerce, its applications are still limited and in the nascent stages. To expand its real-world applications, further development is needed in areas such as interoperability, scalability, and standardization of protocols.

## 8 Open challenges and future directions

The structured literature review has provided insights into existing approaches adapted to improve different aspects of VFL while also pointing out potential research directions by identifying the gaps in the literature, highlighting unresolved issues and noting new developments in VFL. The research directions concluded from the review are discussed as follows:

### 8.1 Model drift

In machine learning, model drift is a critical challenge that impacts the performance of models over time. It can be categorized into two types: concept drift and data drift. Concept drift occurs when the statistical properties of the target variable change, while data drift refers to alterations in the distribution of input data used for training. In VFL, concept drift can affect the model by altering the underlying patterns or relationships the model is trying to learn from different data sources. Since VFL models rely on data from various entities, a change in the underlying concept in one entity can disrupt the overall learning process, leading to inaccurate predictions. Data drift, on the other hand, impacts VFL by changing the statistical properties or distribution of the input data. This can lead to a mismatch between the current data and what the model has learned, causing a decline in model performance as it struggles to make accurate predictions based on the new data characteristics. While considerable research [15, 57, 100] has been conducted on drift handling, notably in HFL, there is a growing need to extend these strategies to VFL. Liang and Chen [80] presented a method, DVFL, designed to adapt to dynamic data distribution changes in VFL through knowledge distillation. However, it still does not address the issue of concept drift. Effective management of drift would involve detection, mitigation, and adaptation of drift as this is crucial for designing fault-tolerant VFL systems. One promising solution lies in incorporating Continual learning [70], which involves the ability of a model to learn continuously, adapt to new data, and retain previously acquired knowledge. By integrating continual learning into VFL, models can become more adaptable to concept and data drift. Moreover, conventional clustering methods have been employed to detect and manage concept drift in machine learning problems. For instance, the Fuzzy Kernel C-Means clustering model [116] has been utilized to handle concept drift in data streams. However, these approaches often lack privacy preservation, which is essential in federated learning settings. Integrating privacy-preserving clustering methods [50, 150] for addressing drift in VFL could be particularly useful, enhancing both drift detection and adaptation securely.

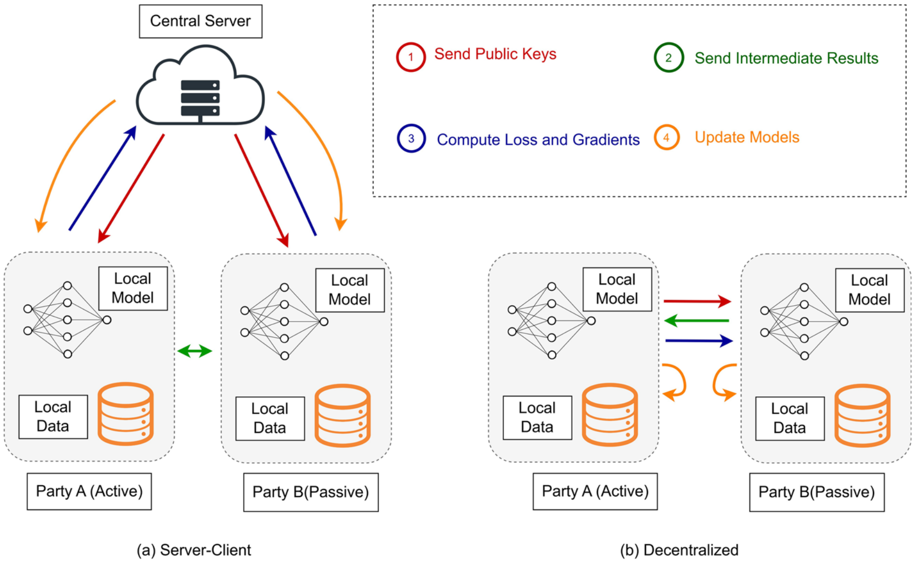


**Fig. 5** Training procedure for server-client (left) and decentralized (right) VFL

## 8.2 Fairness

In the context of ML, fairness can be classified into model fairness and collaborative fairness. Model fairness tackles biases in the data, ensuring that models do not unfairly favor or discriminate against any group. In VFL, if the data from one or more sources is biased, the federated model may develop skewed predictions, favoring certain groups over others. There is a good chance of this happening in VFL due to the diverse and decentralized nature of data sources. To address this issue in VFL, approaches like incorporating regularization techniques [96] and fairness-aware model evaluation [71] can be utilized. Fairness in machine learning is often assessed using bias detection metrics like *Equalized Odds* and *Disparate Impact* [103]. In VFL settings, techniques such as *SMPC* and *HE* can allow participants to collaboratively compute fairness metrics without exposing their private data. Collaborative fairness, on the other hand, focuses on equitable participation, ensuring that all participants in the VFL process are fairly rewarded for their contributions, which is crucial for a balanced and cooperative environment in federated learning setups. Not ensuring collaborative fairness might lead to unequal participation, where certain contributors are undervalued or overburdened. This imbalance can reduce the overall trust in the VFL system from the participant's end (Fig. 5).

## 8.3 Explainability

Explainability is significantly important in vertical federated learning (VFL), as it establishes trust, collaboration, and accountability among participating entities, all while ensuring compliance with strict privacy regulations. However, achieving explainability in VFL is a challenging task due to the inherent decentralization of data, the complexity of federated models, and the prime importance of data privacy. Limited communication between parties, disparate data distributions, and the absence of standardized data structures enhance the challenge. To address these issues, research efforts could focus on developing innovative

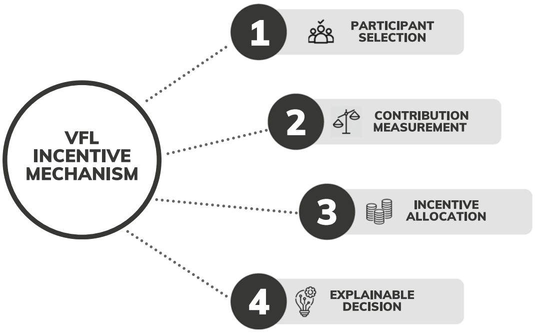


**Fig. 6** Incentive mechanism in VFL: a four step approach

techniques that balance model transparency with data privacy, such as federated explainability methods [72, 162] and privacy-preserving explanations [12] for vertically partitioned data. To further enhance explainability in VFL, research could explore cross-party frameworks that enable shared interpretability while maintaining privacy. By integrating adversarial methods [3] to test and strengthen explanation robustness, and employing federated causal inference [143] to uncover causal relationships in distributed data, VFL systems can achieve a balance between transparency and privacy.

### 8.4 Incentive mechanism

Incentives have an important role in VFL to motivate multiple data parties to come together to train machine learning models while keeping their data decentralized. The right incentives not only encourage participation but also ensure that each party contributes optimally toward the common goal. In the literature, incentive mechanisms in VFL focus on evaluating participants' contributions by using methods such as Shapley values, which fairly assess each participant's value to the collaboration. However, the design of an effective incentive mechanism in VFL is multifaceted and includes several key steps: Participant Selection, Contribution Measurement, Incentive Allocation, and Explainable Decision (Fig. 6).

The first step, Participant Selection in VFL [55] should involve using privacy-preserving techniques to assess data quality of potential participants beforehand, ensuring the selection of those most likely to add value to the federated model. This step helps avoid collaboration with less beneficial participants, aiming for optimal results. Cui et al. [29] proposed an approach for collaboration equilibrium in FL, which facilitates the formation of optimal collaboration coalitions among clients. By using a benefit graph and Pareto optimization. This identifies collaborators who can maximize utility and model performance for each client, ensuring that no client could benefit more by forming different coalitions. In VFL, the utility of collaboration is not solely dependent on data volume or completeness but on the complementarity of the feature sets that each participant brings. This means that determining the maximum achievable utility (MAU) and identifying optimal collaborator sets (OCS) in VFL would require not only understanding individual data contributions but also how these contributions interact in multidimensional and often nonlinear ways. The iterative optimization and the use of benefit graphs in the collaboration equilibrium rely heavily on being able to clearly see how contributions and benefits are distributed among participants. In VFL, this visibility is limited because data must often be encrypted or anonymized. To the best of our knowledge, only two works till now proposed participant selection methods specifically designed for VFL. Jiang et al. [55] proposed using vertically federated Mutual information(MI) estimator to measure mutual information between the labels and features of the participants. A group testing method was further incorporated search for the subset of participants that maximizes this MI. However, this approach requires prior knowledge of how many participants to select which might not be always feasible. To solve this problem, [49] proposed a Joint Mutual Estimator (JMI) for the same purpose as before along with a greedy search algorithm that maximizes the gain of VFL model given by a set of selected participants. But, the scalability of this approach poses doubt, as the greedy search algorithm used is tested only on a limited number of clients (up to 8) and may become computationally expensive with larger participant pools. An important issue not fully addressed is what happens if several participants provide the same or similar data. This could lead to inefficiency. A better approach would be to prioritize participants who offer unique and diverse data, which could make the learning process more effective.

Contribution Measurement in the next step determines the relative importance or impact of each participant's contribution to the performance of the federated model. The cooperative game-theoretic approach, Shapley value has been used commonly for this purpose in existing studies. However, computing Shapley values for participants is computationally expensive. Furthermore, approximation methods intended to reduce this computational burden are not always accurate, leading to potential inaccuracies in measuring contributions. Therefore, exploring other approaches that balance accuracy with computational feasibility still remains a challenge. Thirdly, Incentive Allocation refers to the distribution of rewards, compensation, or other forms of incentives among participants, based on the contributions they have made, as assessed in the Contribution Measurement stage. This is beneficial for encouraging participation, ensuring fairness, and rewarding participants in a way that reflects their contributions to the collaboration. Possible strategies for this purpose can include incorporating game theoretic approaches that deal with rewarding players such as the Nash Bargaining Solution [125] or Nucleolus [93] in VFL. Khan et al. [64] formulated the incentive allocation problem in VFL as a bankruptcy game, where the performance gain achieved by the federated model serves as the "estate," and the claims are derived from the unique contributions of each party to this performance improvement. These claims are satisfied using the Talmud division rule, as Aumann and Maschler [7] proved that this rule inherently computes the Nucleolus of a bankruptcy game without requiring the explicit evaluation of all coalition values, making it both fair and computationally efficient. The final step, Explainable Decision, is vital for the overall integrity and acceptance of the incentive mechanism by the participants. It involves transparently communicating the rationale behind participant selection, contribution assessments, and incentive allocations. Therefore, designing fair and effective incentive mechanism VFL represents a valuable direction for future research.

### 8.5 Dataset availability

Datasets are important for evaluating and benchmarking VFL algorithms or methods, ensuring their effectiveness and applicability to real-world scenarios. Table 7 lists a variety of datasets that are commonly used in VFL studies, spanning different domains. However, in most VFL studies, datasets are often partitioned arbitrarily, lacking standard splitting benchmarks. This approach can lead to unrealistic scenarios, biased samples, and a limited evaluation scope. Such arbitrary partitioning may not accurately represent real-world data distributions and the unique privacy and security challenges in VFL. Moreover, there is a scarcity of datasets supporting multimodality, which hinders research into applying multimodal approaches in VFL.

**Table 7** Datasets used in VFL studies

| Category | Dataset | Data type | No. of features | No. of instances |
|---|---|---|---|---|
| Financial | Income [9] | Tabular | 14 | 48,842 |
| | Bank [95] | Tabular | 16 | 45,211 |
| | Credit Card [152] | Tabular | 24 | 30,000 |
| Medical | MIMIC III [56] | Tabular | Varies by Component | 42,276 |
| | Breast Cancer [134] | Tabular | 30 | 569 |
| | Diabetes Smith et al. [114] | Tabular | 9 | 769 |
| | BHI [94] | Image | N/A | 277,524 |
| | CheXpert [53] | Image | N/A | 65,240 |
| Advertising & marketing | Avazu [58] | Tabular | 24 | 4 M |
| | Criteo [69] | Tabular | 40 | 4.5M |
| Transportation & engineering | Vehicle [32] | Tabular | Not specified | 98,528 |
| | Drive [154] | Tabular | Not specified | 58,509 |
| Multimedia & web | NUSWIDE [27] | Tabular | 84 | 269,648 |
| | MNIST [31] | Image | N/A | 70,000 |
| | ModelNet [138] | Image | N/A | 20,000 |
| | Yahoo Answers [161] | Text | N/A | 1.46M |
| | News20 [62] | Text | N/A | 19,928 |

A recent study [140] addressed this issue and proposed a framework that introduces novel feature-splitting methods for generating synthetic datasets and a new real-world VFL dataset, particularly for image-image scenarios. It also included methods to quantify the importance and correlation in real-world VFL datasets, aligning them with synthetic ones, and conducted comprehensive benchmarks of VFL algorithms across diverse scenarios. Additionally, to address the issue of dataset unavailability, established research communities, task forces, or conferences could take a proactive role by organizing competitions in collaboration with companies across various industries, focusing on realistic VFL challenges. These competitions would encourage the creation of high-quality multimodal datasets that mirror real-world privacy and security constraints. After the competitions, the datasets and benchmarks could be made publicly available which will advance standardized evaluations in VFL problems.

## 9 Conclusion

VFL is increasingly recognized as a practical solution in real-world scenarios, especially by organizations seeking collaborative data analysis while maintaining privacy. Its growing popularity stems from its ability to handle diverse, vertically partitioned datasets across various data parties. This paper presents an extensive review of existing studies in VFL, summarizing them in a categorized manner and outlining the unexplored challenges and potential research paths. By identifying these gaps, we aim to pave the way for future advancements in VFL. This review aspires to serve as a valuable resource for inspiring researchers, both advanced and new to this domain.

**Author Contributions** Conceptualization, A.K., M.t.T. and A.W.; Data curation, A.K.; Investigation, A.K.; Methodology, A.K., M.t.T. and A.W.; Supervision, M.t.T. and A.W.; Writing-original draft, A.K.; Writing-review and editing, M.t.T. and A.W.

**Funding** Not applicable.

**Data Availability** No associated data in data repository.

## Declarations

**Conflict of interest** The authors declare no conflict of interest.

**Ethical and informed consent for data used** Not Applicable.

## References

1. Abadi A, Doyle B, Gini F, et al (2024) Starlit: Privacy-preserving federated learning to enhance financial fraud detection. arXiv preprint arXiv:2401.10765

2. Agarwal A, Dudík M, Kale S, et al (2012) Contextual bandit learning with predictable rewards. In: Proceedings of the 15th international conference on artificial intelligence and statistics, PMLR, pp 19–26
3. Agrawal N, Pendharkar I, Shroff J et al (2024) A-XAI: adversarial machine learning for trustable explainability. AI and Ethics 4:1143–1174
4. Albrecht JP (2016) How the GDPR will change the world. Eur Data Protect Law Rev 2:287–289
5. Aono Y, Hayashi T, Wang L et al (2017) Privacy-preserving deep learning via additively homomorphic encryption. IEEE Trans Inf Forensics Secur 13(5):1333–1345
6. Armitage A, Keeble-Allen D (2008) Undertaking a structured literature review or structuring a literature review: Tales from the field. In: Proceedings of the 7th European conference on research methodology for business and management studies: ECRM2008, Regent's College, London, 35
7. Aumann RJ, Maschler M (1985) Game theoretic analysis of a bankruptcy problem from the talmud. J Econom Theory 36(2):195–213
8. Baidu (2021) GitHub - PaddlePaddle/PaddleFL: Federated Deep Learning in PaddlePaddle — github.com. URL github.com/PaddlePaddle/PaddleFL, [Accessed 16-07-2024]
9. Becker B, Kohavi R (1996) Adult Dataset. UCI machine learning repository, https://doi.org/10.24432/C5XW20
10. Beutel DJ, Topal T, Mathur A, et al (2020) Flower: a friendly federated learning research framework. arXiv preprint arXiv:2007.14390
11. Bhardwaj R, Nambiar AR, Dutta D (2017) A study of machine learning in healthcare. In: Proceedings of the 41st annual computer software and applications conference (COMPSAC), IEEE, 236–241
12. Bogdanova A, Imakura A, Sakurai T (2023) DC-SHAP method for consistent explainability in privacy-preserving distributed machine learning. Human-Centric Intell Syst 3(3):197–210
13. Bonawitz K, Ivanov V, Kreuter B, et al (2017) Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security, 1175–1191
14. Buddhavarapu P, Knox A, Mohassel P, et al (2020) Private matching for compute. Cryptology ePrint Archive, Paper 2020/599
15. Canonaco G, Bergamasco A, Mongelluzzo A, et al (2021) Adaptive federated learning in presence of concept drift. In: Proceedings of the 2021 international joint conference on neural networks (IJCNN), IEEE, 1–7
16. Cao Z, Liang Z, Wu B, et al (2023) Privacy matters: vertical federated linear contextual bandits for privacy protected recommendation. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 154–166
17. Castiglia T, Zhou Y, Wang S, et al (2023) LESS-VFL: Communication-efficient feature selection for vertical federated learning. In: Proceedings of the 40th international conference on machine learning, proceedings of machine learning research, vol 202. PMLR, 3757–3781
18. Castiglia TJ, Das A, Wang S, et al (2022) Compressed-VFL: Communication-efficient learning with vertically partitioned data. In: Proceedings of the 39th international conference on machine learning, PMLR, 2738–2766
19. Cha D, Sung M, Park YR, et al (2021) Implementing vertical federated learning using autoencoders: Practical application, generalizability, and utility study. JMIR Med Inform, 9(6):e26,598
20. Chase M, Miao P (2020) Private set intersection in the internet setting from lightweight oblivious PRF. In: Proceedings of the 40th annual international cryptology conference, Springer, 34–63
21. Chen P, Du X, Lu Z et al (2022) EVFL: an explainable vertical federated learning for data-oriented artificial intelligence systems. J Syst Architect 126(102):474
22. Chen T, Jin X, Sun Y, et al (2020) VAFL: a method of vertical asynchronous federated learning. arXiv preprint arXiv:2007.06081
23. Cheng K, Fan T, Jin Y et al (2021) Secureboost: a lossless federated learning framework. IEEE Intell Syst 36(6):87–98
24. Cheng Y, Liu Y, Chen T et al (2020) Federated learning for privacy-preserving AI. Commun ACM 63(12):33–36
25. Cheung Ym, Lou J, Yu F (2021) Vertical federated principal component analysis on feature-wise distributed data. In: Proceedings of the 22nd international conference on web information systems engineering, Springer, 173–188
26. Cheung Ym, Jiang J, Yu F, et al (2022) Vertical federated principal component analysis and its kernel extension on feature-wise distributed data. arXiv preprint arXiv:2203.01752
27. Chua TS, Tang J, Hong R, et al (2009) NUS-WIDE: a real-world web image database from National University of Singapore. In: Proceedings of the ACM international conference on image and video retrieval, pp 1–9

28. Cristofaro ED, Tsudik G (2010) Practical private set intersection protocols with linear complexity. In: Proceedings of the 14th international conference on financial cryptography and data security, Springer, 143–159
29. Cui S, Liang J, Pan W, et al (2022) Collaboration equilibrium in federated learning. In: Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining, pp 241–251
30. Dai M, Xu A, Huang Q et al (2021) Vertical federated DNN training. Phys Commun 49(101):465
31. Deng L (2012) The MNIST database of handwritten digit images for machine learning research [Best of the Web]. IEEE Signal Process Mag 29(6):141–142
32. Duarte MF, Hu YH (2004) Vehicle classification in distributed sensor networks. J Parallel Distrib Comput 64(7):826–838
33. Efe Y (2021) A vertical federated learning method for multi-institutional credit scoring: MICS. arXiv preprint arXiv:2111.09038
34. Fan Z, Fang H, Zhou Z, et al (2022) Fair and efficient contribution valuation for vertical federated learning. arXiv preprint arXiv:2201.02658
35. Feng S (2022) Vertical federated learning-based feature selection with non-overlapping sample utilization. Expert Syst Appl 208(118):097
36. Feng S, Yu H (2020) Multi-participant multi-class vertical federated learning. arXiv preprint arXiv:2001.11154
37. Freedman MJ, Nissim K, Pinkas B (2004) Efficient private matching and set intersection. In: Proceedings of the 23rd international conference on the theory and applications of cryptographic techniques, Springer, 1–19
38. Fu C, Zhang X, Ji S, et al (2022a) Label inference attacks against vertical federated learning. In: Proceedings of the 31st USENIX Security Symposium, Security 2022, USENIX Association, 1397–1414
39. Fu F, Miao X, Jiang J et al (2022) Towards communication-efficient vertical federated learning training via cache-enabled local updates. Proc VLDB Endowment 15(10):2111–2120
40. Gao D, Liu Y, Huang A, et al (2019) Privacy-preserving heterogeneous federated transfer learning. In: Proceedings of the 2019 IEEE international conference on big data (Big Data), IEEE, 2552–2559
41. Gascón A, Schoppmann P, Balle B et al (2017) Privacy-preserving distributed linear regression on high-dimensional data. Proc Privacy Enhanc Technol 4:345–364
42. Ghazi B, Golowich N, Kumar R et al (2021) Deep learning with label differential privacy. Adv Neural Inf Process Syst 34:27,131-27,145
43. Goodell JW, Kumar S, Lim WM et al (2021) Artificial intelligence and machine learning in finance: identifying foundations, themes, and research clusters from bibliometric analysis. J Behav Exp Financ 32(100):577
44. Gu B, Xu A, Huo Z et al (2021) Privacy-preserving asynchronous vertical federated learning algorithms for multiparty collaborative learning. IEEE Trans Neural Netw Learn Syst 33(11):6103–6115
45. Guo L, Wei Q, Chenyu H (2021) Research on flight delay prediction based on horizontal and vertical federated learning framework. In: Proceedings of the international conference on civil aviation safety and information technology (ICCASIT), IEEE, 38–44
46. Hardy S, Henecka W, Ivey-Law H, et al (2017) Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. arXiv preprint arXiv:1711.10677
47. He C, Li S, So J, et al (2020) FedML: A research library and benchmark for federated machine learning. arXiv preprint arXiv:2007.13518
48. He D, Du R, Zhu S et al (2021) Secure logistic regression for vertical federated learning. IEEE Internet Comput 26(2):61–68
49. Huang J, Zhang L, Li A, et al (2023) Adaptive and efficient participant selection in vertical federated learning. In: Proceedings of the 19th international conference on mobility, sensing and networking (MSN), IEEE, 455–462
50. Huang L, Li Z, Sun J et al (2022) Coresets for vertical federated learning: regularized linear regression and $k$-means clustering. Adv Neural Inf Process Syst 35:29,566-29,581
51. Huang Y, Evans D, Katz J (2012) Private set intersection: are garbled circuits better than custom protocols? In: Proceedings of the network and distributed system security symposium
52. Ion M, Kreuter B, Nergiz AE, et al (2020) On deploying secure computing: private intersection-sum-with-cardinality. In: Proceedings of the 5th European symposium on security and privacy (EuroS&P), IEEE, 370–389
53. Irvin J, Rajpurkar P, Ko M, et al (2019) Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: Proceedings of the AAAI conference on artificial intelligence, pp 590–597
54. Islam TU, Mohammed N, Alhadidi D (2024) Privacy preserving vertical distributed learning for health data. J Surv Secur Saf 5(1):1–18

55. Jiang J, Burkhalter L, Fangcheng F, et al (2022) VF-PS: How to Select Important Participants in Vertical Federated Learning, Efficiently and Securely? In: Proceedings of the 36th international conference on neural information processing systems, 2088 – 2101
56. Johnson AE, Pollard TJ, Shen L et al (2016) MIMIC-III, a freely accessible critical care database. Sci Data 3(1):1–9
57. Jothimurugesan E, Hsieh K, Wang J, et al (2023) Federated learning under distributed concept drift. In: Proceedings of the 26th international conference on artificial intelligence and statistics, PMLR, 5834–5853
58. Kaggle (2014) Click-Through Rate Prediction — kaggle.com. URL www.kaggle.com/c/avazu-ctr-prediction, [Accessed 16-07-2024]
59. Kairouz P, McMahan HB, Avent B et al (2021) Advances and open problems in federated learning. Found Trends Regist Mach Learn 14(1–2):1–210
60. Kang Y, He Y, Luo J et al (2022) Privacy-preserving federated adversarial domain adaptation over feature groups for interpretability. IEEE Trans Big Data 10:879–890
61. Kang Y, Liu Y, Liang X (2022) Fedcvt: semi-supervised vertical federated learning with cross-view training. ACM Trans Intell Syst Technol (TIST) 13(4):1–16
62. Keerthi SS, DeCoste D, Joachims T (2005) A modified finite newton method for fast solution of large scale linear SVMs. J Mach Learn Res 6(3):341–361
63. Khan A, ten Thij M, Wilbik A (2022) Communication-efficient vertical federated learning. Algorithms 15(8):273
64. Khan A, ten Thij M, Thuijsman F, et al (2024) Using the nucleolus for incentive allocation in vertical federated learning. In: Proceedings of the IEEE 2nd international conference on federated learning technologies and applications [In Press]
65. Khodaparast F, Sheikhalishahi M, Haghighi H, et al (2018) Privacy preserving random decision tree classification over horizontally and vertically partitioned data. In: Proceedings of the 16th international conference on dependable, autonomic and secure computing, 16th international conference on pervasive intelligence and computing, 4th international conference on big data intelligence and computing and cyber science and technology congress (DASC/PiCom/DataCom/CyberSciTech), IEEE, 600–607
66. Kikuchi H, Hamanaga C, Yasunaga H et al (2018) Privacy-preserving multiple linear regression of vertically partitioned real medical datasets. J Inf Process 26:638–647
67. Kitchenham BA (2012) Systematic review in software engineering: where we are and where we should be going. In: Proceedings of the 2nd international workshop on Evidential assessment of software technologies, 1–2
68. Knott B, Venkataraman S, Hannun A et al (2021) Crypten: secure multi-party computation meets machine learning. Adv Neural Inf Process Syst 34:4961–4973
69. Lab CA (2014) Kaggle display advertising challenge dataset. URL ailab.criteo.com/ressources/, accessed: [16-07-2024]
70. Le J, Lei X, Mu N et al (2021) Federated continuous learning with broad network architecture. IEEE Trans Cybern 51(8):3874–3888
71. Le Quy T, Roy A, Iosifidis V et al (2022) A survey on datasets for fairness-aware machine learning. Wiley Interdiscip Rev: Data Min Knowl Discov 12(3):e1452
72. Li A, Liu R, Hu M, et al (2023a) Towards interpretable federated learning. arXiv preprint arXiv:2302.13473
73. Li A, Peng H, Zhang L, et al (2023b) FedSDG-FS: Efficient and secure feature selection for vertical federated learning. arXiv preprint arXiv:2302.10417
74. Li D, Wang J, Kong L, et al (2022a) A nearest neighbor under-sampling strategy for vertical federated learning in financial domain. In: Proceedings of the 2022 ACM workshop on information hiding and multimedia security, 123–128
75. Li M, Chen Y, Wang Y, et al (2020) Efficient asynchronous vertical federated learning via gradient prediction and double-end sparse compression. In: Proceedings of the 16th international conference on control, automation, robotics and vision (ICARCV), IEEE, 291–296
76. Li Q, Thapa C, Ong L, et al (2023c) Vertical federated learning: taxonomies, threats, and prospects. arXiv preprint arXiv:2302.01550
77. Li Q, ZHAOMIN W, Cai Y, et al (2023d) Fedtree: A federated learning system for trees. In: Proceedings of sixth conference machine learning and systems
78. Li W, Xia Q, Cheng H, et al (2022b) Vertical semi-federated learning for efficient online advertising. arXiv preprint arXiv:2209.15635
79. Liakos KG, Busato P, Moshou D et al (2018) Machine learning in agriculture: a review. Sensors 18(8):2674

80. Liang Y, Chen Y (2021) DVFL: a vertical federated learning method for dynamic data. arXiv preprint arXiv:2111.03341
81. Liang Y, Liu Z, Song Y, et al (2021) A methodology of trusted data sharing across telecom and finance sector under china's data security policy. In: Proceedings of the ieee international conference on big data, IEEE, 5406–5412
82. Lin Y, Han S, Mao H, et al (2017) Deep gradient compression: reducing the communication bandwidth for distributed training. arXiv preprint arXiv:1712.01887
83. Liu C, Zhou Z, Shi Y, et al (2021a) Achieving model fairness in vertical federated learning. arXiv preprint arXiv:2109.08344
84. Liu F, Wu X, Ge S, et al (2020) Federated learning for vision-and-language grounding problems. In: Proceedings of the AAAI conference on artificial intelligence, 11,572–11,579
85. Liu J, Xu H, Wang L et al (2021) Adaptive asynchronous federated learning in resource-constrained edge computing. IEEE Trans Mob Comput 22(2):674–690
86. Liu Y, Fan T, Chen T et al (2021) FATE: an industrial grade platform for collaborative learning with data protection. J Mach Learn Res 22(1):10,320-10,325
87. Liu Y, Zhang X, Kang Y et al (2022) FedBCD: a communication-efficient collaborative learning framework for distributed features. IEEE Trans Signal Process 70:4277–4290
88. Liu Y, Kang Y, Zou T et al (2024) Vertical federated learning: concepts, advances, and challenges. IEEE Trans Knowl Data Eng 36(7):3615–3634
89. Lu L, Ding N (2020) Multi-party private set intersection in vertical federated learning. In: Proceedings of the 19th International conference on trust, security and privacy in computing and communications (TrustCom), IEEE, 707–714
90. Luo X, Wu Y, Xiao X, et al (2021) Feature inference attack on model predictions in vertical federated learning. In: Proceedings of the 37th international conference on data engineering (ICDE), IEEE, 181–192
91. Ma C, Li J, Ding M et al (2020) On safeguarding privacy and security in the framework of federated learning. IEEE Netw 34(4):242–248
92. Mai P, Pang Y (2023) Vertical federated graph neural network for recommender system. In: Proceedings of the 40th international conference on machine learning, PMLR, 23,516–23,535
93. Maschler M (1992) The bargaining set, kernel, and nucleolus. Handbook Game Theory Econom Appl 1:591–667
94. Mooney P (2017) Breast Histopathology Images. URL www.kaggle.com/datasets/paultimothymooney/breast-histopathology-images, Accessed on 16 July 2024
95. Moro S, Rita P, Cortez P (2012) Bank marketing dataset. UCI machine learning repository, https://doi.org/10.24432/C5K306
96. Mosteiro P, Kuiper J, Masthoff J et al (2022) Bias discovery in machine learning models for mental health. Information 13(5):237
97. Mugunthan V, Goyal P, Kagal L (2021) Multi-VFL: a vertical federated learning system for multiple data and label owners. arXiv preprint arXiv:2106.05468
98. Nguyen Duy D, Affenzeller M, Nikzad-Langerodi R (2023) Towards vertical privacy-preserving symbolic regression via secure multiparty computation. In: Proceedings of the companion conference on genetic and evolutionary computation, 2420–2428
99. Ou W, Zeng J, Guo Z et al (2020) A homomorphic-encryption-based vertical federated learning scheme for rick management. Comput Sci Inf Syst 17(3):819–834
100. Ozfatura E, Ozfatura K, Gündüz D (2021) Fedadc: Accelerated federated learning with drift control. In: Proceedings of the IEEE international symposium on information theory (ISIT), IEEE, 467–472
101. Paillier P (1999) Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of the international conference on the theory and applications of cryptographic techniques, Springer, 223–238
102. Pechyony D, Vapnik V (2010) On the theory of learning with privileged information. In: Proceedings of the 23rd international conference on neural information processing systems , 2, 1894-1902
103. Pessach D, Shmueli E (2022) A review on fairness in machine learning. ACM Comput Surv (CSUR) 55(3):1–44
104. Qi T, Wu F, Wu C et al (2022) FairVFL: a fair vertical federated learning framework with contrastive adversarial learning. Adv Neural Inf Process Syst 35:7852–7865
105. Qiu P, Zhang X, Ji S, et al (2022) All you need is hashing: defending against data reconstruction attack in vertical federated learning. arXiv preprint arXiv:2212.00325
106. Ratadiya P, Asawa K, Nikhal O (2020) A decentralized aggregation mechanism for training deep learning models using smart contract system for bank loan prediction. arXiv preprint arXiv:2011.10981

107. Ren Z, Yang L, Chen K (2022) Improving availability of vertical federated learning: relaxing inference on non-overlapping data. ACM Trans Intell Syst Technol (TIST) 13:1–20
108. Romanini D, Hall AJ, Papadopoulos P, et al (2021) Pyvertical: a vertical federated learning framework for multi-headed splitnn. arXiv preprint arXiv:2104.00489
109. Roth AE (1988) The Shapley value: essays in honor of Lloyd S. Cambridge University Press, Shapley
110. Roth HR, Cheng Y, Wen Y, et al (2022) NVIDIA FLARE: federated learning from simulation to real-world. arXiv preprint arXiv:2210.13291
111. Sha T, Yu X, Shi Z, et al (2021) Feature map transfer: Vertical federated learning for cnn models. In: Proceedings of the international conference on data mining and big data, Springer, 37–44
112. Shi H, Jiang Y, Yu H, et al (2022) MVFLS: multi-participant vertical federated learning based on secret sharing. In: International workshop on trustable, verifiable and auditable federated learning, 1–9
113. Shokri R, Shmatikov V (2015) Privacy-preserving deep learning. In: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, 1310–1321
114. Smith JW, Everhart JE, Dickson W, et al (1988) Using the adap learning algorithm to forecast the onset of diabetes mellitus. In: Proceedings of the annual symposium on computer application in medical care, American Medical Informatics Association, p 261
115. Song T, Tong Y, Wei S (2019) Profit allocation for federated learning. In: Proceedings of the international conference on big data (Big Data), IEEE, 2577–2586
116. Song Y, Zhang G, Lu J, et al (2017) A fuzzy kernel c-means clustering model for handling concept drift in regression. In: Proceedings of the international conference on fuzzy systems (FUZZ-IEEE), 1–6
117. Streeton R, Cooke M, Campbell J (2004) Researching the researchers: using a snowballing technique. Nurse Res 12(1):35–47
118. Sun C, Ippel L, Van Soest J, et al (2019) A privacy-preserving infrastructure for analyzing personal health data in a vertically partitioned scenario. MEDINFO 2019: Health and Wellbeing e-Networks for All 264:373–377
119. Sun H, Wang Z, Huang Y, et al (2022a) Privacy-preserving vertical federated logistic regression without trusted third-party coordinator. In: Proceedings of the 6th international conference on machine learning and soft computing, 132–138
120. Sun J, Yao Y, Gao W, et al (2021a) Defending against reconstruction attack in vertical federated learning. arXiv preprint arXiv:2107.09898
121. Sun J, Yang X, Yao Y, et al (2022b) Label leakage and protection from forward embedding in vertical federated learning. arXiv preprint arXiv:2203.01451
122. Sun J, Yang X, Yao Y, et al (2022c) Differentially private AUC computation in vertical federated learning. arXiv preprint arXiv:2205.12412
123. Sun W, Chen Y, Yang X, et al (2021b) FedIO: bridge inner-and outer-hospital information for perioperative complications prognostic prediction via federated learning. In: Proceedings of the international conference on bioinformatics and biomedicine (BIBM), IEEE, 3215–3221
124. Tang F, Liang S, Ling G et al (2023) IHVFL: a privacy-enhanced intention-hiding vertical federated learning framework for medical data. Cybersecurity 6(1):37
125. Van Damme E (1986) The Nash bargaining solution is optimal. J Econom Theory 38(1):78–100
126. Wang C, Liang J, Huang M, et al (2020a) Hybrid differentially private federated learning on vertically partitioned data. arXiv preprint arXiv:2009.02763
127. Wang G, Dang CX, Zhou Z (2019) Measure contribution of participants in federated learning. In: Proceedings of the IEEE international conference on big data (Big Data), IEEE, 2597–2604
128. Wang J, Zhang L, Li A, et al (2022a) Efficient participant contribution evaluation for horizontal and vertical federated learning. In: Proceedings of the 38th international conference on data engineering (ICDE), IEEE, 911–923
129. Wang R, Ersoy O, Zhu H et al (2022) FEVERLESS: fast and secure vertical federated learning based on XGBoost for decentralized labels. IEEE Trans Big Data 10(6):1001–1015
130. Wang T, Rausch J, Zhang C, et al (2020b) A principled approach to data valuation for federated learning. Federated learning: privacy and incentive, 153–167
131. Wang Y, Lv Q, Zhao M et al (2023) Beyond model splitting: preventing label inference attacks in vertical federated learning with dispersed training. World Wide Web 26:2691–2707
132. Wei Q, Li Q, Zhou Z et al (2021) Privacy-preserving two-parties logistic regression on vertically partitioned data using asynchronous gradient sharing. Peer-to-Peer Netw Appl 14(3):1379–1387
133. WenJie S, Xuan S (2021) Vertical federated learning based on dfp and bfgs. arXiv preprint arXiv:2101.09428
134. Wolberg W, Mangasarian O, Street N, et al (1995) Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository, https://doi.org/10.24432/C5DW2B

135. Worm D, Kamphorst B, Rooijakkers T, et al (2020) CONVINCED—Enabling privacy-preserving survival analyses using Multi-Party Computation. Tech. Rep. 880221, TNO, URL https://resolver.tno.nl/uuid:1c4885d6-8cf3-4443-b952-e887e1b41207
136. Wu C, Wu F, Wang X, et al (2021) Fairness-aware news recommendation with decomposed adversarial learning. In: Proceedings of the AAAI conference on artificial intelligence, 4462–4469
137. Wu Y, Cai S, Xiao X, et al (2020) Privacy preserving vertical federated learning for tree-based models. arXiv preprint arXiv:2008.06170
138. Wu Z, Song S, Khosla A, et al (2015) 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 1912–1920
139. Wu Z, Li Q, He B (2022) Practical vertical federated learning with unsupervised representation learning. IEEE Trans Big Data 10:864–878
140. Wu Z, Hou J, He B (2023) Vertibench: Advancing feature distribution diversity in vertical federated learning benchmarks. arXiv preprint arXiv:2307.02040
141. Xie C, Chen PY, Zhang C, et al (2022a) Improving privacy-preserving vertical federated learning by efficient communication with ADMM. arXiv preprint arXiv:2207.10226
142. Xie Y, Wang Z, Gao D, et al (2022b) FederatedScope: a flexible federated learning platform for heterogeneity. arXiv preprint arXiv:2204.05011
143. Xiong R, Koenecke A, Powell M et al (2023) Federated causal inference in heterogeneous observational data. Stat Med 42(24):4418–4439
144. Xu D, Yuan S, Wu X (2021a) Achieving differential privacy in vertically partitioned multiparty learning. In: Proceedings of the IEEE international conference on big data (Big Data), IEEE, 5474–5483
145. Xu R, Baracaldo N, Zhou Y, et al (2021b) FedV: Privacy-preserving federated learning over vertically partitioned data. In: Proceedings of the 14th ACM workshop on artificial intelligence and security, 181–192
146. Yang K, Fan T, Chen T, et al (2019a) A quasi-newton method based vertical federated learning framework for logistic regression. arXiv preprint arXiv:1912.00513
147. Yang K, Song Z, Zhang Y, et al (2021) Model optimization method based on vertical federated learning. In: Proceedings of the international symposium on circuits and systems (ISCAS), IEEE, 1–5
148. Yang L, Chai D, Zhang J, et al (2023) A survey on vertical federated learning: from a layered perspective. arXiv preprint arXiv:2304.01829
149. Yang S, Ren B, Zhou X, et al (2019b) Parallel distributed logistic regression for vertical federated learning without third-party coordinator. arXiv preprint arXiv:1911.09824
150. Yassen MA, Muhammed LAN (2024) Vertical federated learning with k-means and k-mode. In: Proceedings of American institute of physics conference, AIP Publishing
151. Ye P, Jiang Z, Wang W, et al (2022) Feature reconstruction attacks and countermeasures of DNN training in vertical federated learning. arXiv preprint arXiv:2210.06771
152. Yeh IC (2016) Default of Credit Card Clients. UCI Machine Learning Repository, https://doi.org/10.24432/C55S3H
153. Yi X, Paulet R, Bertino E (2014) Homomorphic encryption. Springer, Cham
154. Yuan J, Zheng Y, Zhang C, et al (2010) T-drive: driving directions based on taxi trajectories. In: Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems, 99–108
155. Zhang J, Jiang Y (2021) A vertical federation recommendation method based on clustering and latent factor model. In: Proceedings of the international conference on electronic information engineering and computer science (EIECS), IEEE, 362–366
156. Zhang J, Jiang Y (2022) A data augmentation method for vertical federated learning. Wirel Commun Mob Comput 2022:1–16
157. Zhang J, Guo S, Qu Z et al (2022) Adaptive vertical federated learning on unbalanced features. IEEE Trans Parallel Distrib Syst 33(12):4006–4018
158. Zhang Q, Gu B, Deng C, et al (2021a) Asysqn: Faster vertical federated learning algorithms with better computation resource utilization. In: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, 3917–3927
159. Zhang Q, Gu B, Deng C, et al (2021b) Secure bilevel asynchronous vertical federated learning with backward updating. In: Proceedings of the AAAI conference on artificial intelligence, 10,896–10,904
160. Zhang R, Li H, Hao M, et al (2022b) Secure feature selection for vertical federated learning in ehealth systems. In: Proceedings of the international conference on communications, IEEE, 1257–1262
161. Zhang X, Zhao J, LeCun Y (2015) Character-level convolutional networks for text classification. Adv Neural Inf Process Syst, 28
162. Zhang Y, Yu H (2023) LR-XFL: logical reasoning-based explainable federated learning. arXiv preprint arXiv:2308.12681

163. Zhang Y, Wu Q, Shikh-Bahaei M (2020) Vertical federated learning based privacy-preserving cooperative sensing in cognitive radio networks. In: Proceedings of the 2020 IEEE Globecom workshops, IEEE, 1–6
164. Zhang Y, Hu Y, Gao X et al (2022) An embedded vertical-federated feature selection algorithm based on particle swarm optimisation. CAAI Trans Intell Technol 8:734–754
165. Zhang Z, Li X, Yang S (2022d) Data pricing in vertical federated learning. In: Proceedings of the 2022 IEEE/CIC international conference on communications in China (ICCC), 932–937
166. Zhao D, Yao M, Wang W, et al (2022) NTP-VFL-A New Scheme for Non-3rd Party Vertical Federated Learning. In: Proceedings of the 2022 14th international conference on machine learning and computing (ICMLC), 134–139
167. Zhou J, Chen C, Zheng L, et al (2020) Vertically federated graph neural network for privacy-preserving node classification. arXiv preprint arXiv:2005.11903
168. Zhou X, Liang W, Ma J et al (2022) 2D federated learning for personalized human activity recognition in cyber-physical-social systems. IEEE Trans Netw Sci Eng 9(6):3934–3944
169. Zhu H, Wang R, Jin Y et al (2021) PIVODL: privacy-preserving vertical federated learning over distributed labels. IEEE Trans Artif Intell 4:988–1001
170. Zhu H, Xu J, Liu S et al (2021) Federated learning on non-iid data: a survey. Neurocomputing 465:371–390
171. Zhu L, Liu Z, Han S (2019) Deep leakage from gradients. Adv Neural Inf Process Syst, 32
172. Ziller A, Trask A, Lopardo A, et al (2021) PySyft: A library for easy federated learning, Studies in Computational Intelligence, vol 965, Springer International Publishing, 111–139
173. Zou T, Liu Y, Kang Y et al (2022) Defending batch-level label inference and replacement attacks in vertical federated learning. IEEE Trans Big Data 10:1016–1027

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Afsana Khan** is a PhD candidate at the Department of Advanced Computing Sciences at Maastricht University. Her research focuses on developing proof-of-concept solutions to improve communication and learning efficiency in vertical federated learning (VFL), a privacy-preserving machine learning approach that enables collaborative training of models without sharing raw data. Previously, she earned her MSc in Computer Science with a specialization in distributed learning from the University of Tartu, Estonia. Her broader research interests include privacy-preserving machine learning, federated learning and data fusion.

**Marijn ten Thij** is an Assistant Professor at the Department of Cognitive Science and Artificial Intelligence at Tilburg University. Marijn obtained a MSc in Applied Mathematics at the University of Twente and a PhD in Mathematics (Business Analytics) at the Vrije Universiteit Amsterdam. Marijn's research interests involve the usage of mathematical modelling to study and mimic human behavior through data obtained from social media. His current work is at the intersection of the fields Complex Networks, Computational Social Science, and Data Science.

**Anna Wilbik** is Professor in Data Fusion and Intelligent Interaction at Department of Advanced Computing Sciences, Maastricht University and PI at Brightlands Institute for Smart Society, Maastricht University. Anna Wilbik received her PhD (with honors) in Computer Science from the Systems Research Institute, Polish Academy of Science, Warsaw, Poland, in 2010. She was a Post-doctoral Fellow at University of Missouri, Columbia, USA and later an Assistant Professor at Eindhoven University of Technology, The Netherlands. Her research interests aim at bridging the gap between the meaning of data and human understanding in complex application environments, where data can be of various natures.