International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Federated Learning: Collaborative Machine Learning on Disseminated Data

Shreyash Borole [(1)], Tejas Gorivale [(2)]

Department of Computer Science, Ramrao Adik Institute of Technology, Navi Mumbai
Email: shreyashborole18@gmail.com, tejasgorivale14@gmail.com

ABSTRACT:

Collaborative machine learning on distributed data has emerged as a crucial paradigm in the field of artificial intelligence and data science. In an era marked by the proliferation of decentralized data sources and privacy concerns, this approach offers a promising solution to harness the collective intelligence of data while respecting data sovereignty and confidentiality.

This abstract provides an overview of the key aspects and challenges associated with collaborative machine learning on distributed data. We explore the motivations behind this paradigm, highlighting the need for decentralized learning in scenarios involving geographically dispersed data repositories, sensitive information, or regulatory constraints.

The central theme revolves around enabling multiple parties or entities to jointly train machine learning models without sharing their raw data. Techniques such as federated learning, secure multi-party computation, and blockchain-based data sharing are discussed as foundational building blocks for collaborative machine learning. These approaches empower organizations and individuals to pool their knowledge while preserving data privacy and security.

Challenges related to data heterogeneity, communication overhead, model aggregation, and trust establishment are examined in depth. We also delve into the potential applications of collaborative machine learning, spanning various domains such as healthcare, finance, IoT, and more.

Keywords: Collaborative Machine Learning, Federated Learning, Distributed Data, Decentralized Machine Learning, Privacy-Preserving Machine Learning, Secure Multi-Party Computation (SMPC), Data Federations, Edge Computing, Horizontal Federated Learning, Vertical Federated Learning, Model Aggregation, Data Sharing Protocols, Differential Privacy, Secure Aggregation, Decentralized Data Governance, Consensus Algorithms, Trustworthy AI, Blockchain for Machine Learning, Secure Enclaves.

1. Introduction:

Collaborative Machine Learning on Distributed Data is an emerging field in the realm of machine learning and artificial intelligence (AI) that addresses the challenges of training and deploying models when data is distributed across multiple sources or locations. This collaborative approach is essential in scenarios where data privacy, security, or regulatory constraints prevent centralizing data in a single location.

Collaborative Machine Learning on Distributed Data is a cutting-edge field that addresses the challenges of privacy, security, and data distribution in the context of machine learning. It enables organizations and entities to work together effectively to leverage their data for building powerful AI models while respecting privacy and regulatory constraints.

In traditional machine learning, data is typically collected, stored, and processed in a centralized manner. However, this centralized approach can raise concerns about privacy, data ownership, and data movement. Collaborative Machine Learning on Distributed Data aims to overcome these challenges by enabling multiple parties or organizations to work together to train and deploy machine learning models while keeping their data decentralized.

1.1 Key components and concepts of Collaborative Machine Learning on Distributed Data include:

1. Federated Learning: Federated Learning is a prominent technique in collaborative machine learning. It allows multiple parties, such as individual devices, organizations, or edge nodes, to collaboratively train a global machine learning model without sharing their raw data. Instead, model updates are exchanged between the parties, and a global model is aggregated from these updates. This approach preserves data privacy and reduces the risks associated with data sharing.

2. Distributed Data Sources: In collaborative settings, data can be distributed across various sources, such as different companies, geographic locations, or devices. These sources can be connected through secure communication channels to facilitate the sharing of model updates without exposing the raw data.

3. Privacy-Preserving Techniques: Collaborative Machine Learning often employs privacy-preserving techniques like differential privacy, secure multi-party computation, and homomorphic encryption to ensure that sensitive information remains confidential while contributing to model training.

4. Decentralized Model Deployment: After training, decentralized models can be deployed to different locations or devices, allowing predictions to be made locally without transmitting sensitive data to a central server. This is particularly valuable in edge computing scenarios.

5. Regulatory Compliance: Collaborative Machine Learning must adhere to various regulatory requirements, such as GDPR in Europe or HIPAA in the United States. Techniques like data anonymization and consent management are crucial for compliance.

6. Data Governance and Trust: Establishing trust among collaborators is essential in collaborative machine learning. Clear data governance policies, auditing mechanisms, and trust frameworks help build confidence among participants.

1.2 Applications of Collaborative Machine Learning on Distributed Data are numerous and diverse. They include:

1. **Healthcare:** Collaboration between hospitals, clinics, and researchers to train predictive models without sharing patient data.
2. **Finance:** Banks and financial institutions can collaborate on fraud detection and risk assessment models without exposing customer information.
3. **IoT:** Edge devices can collaborate to improve local decision-making, such as autonomous vehicles sharing information about road conditions.
4. **Cross-Organization Research:** Academic institutions and research organizations can pool their data resources for more robust scientific studies while preserving data privacy.

2. Material and methods on Collaborative machine learning on distributed data

Collaborative machine learning on distributed data refers to the process of training machine learning models on data that is distributed across multiple sources or locations. This approach is commonly used in situations where data cannot be centralized due to privacy, security, or scalability concerns. Below, I'll outline the typical materials and methods used in collaborative machine learning on distributed data:

2.1 Materials:

1. Distributed Data Sources: Identify and describe the data sources that will be used for collaborative machine learning. These can include multiple databases, edge devices, sensors, or remote servers.

2. Hardware Infrastructure: Specify the hardware components, including CPUs, GPUs, TPUs, and network infrastructure, that will be used to facilitate distributed training.

3. Software Tools and Frameworks: Mention the software tools and frameworks you'll use for collaborative machine learning. Common choices include TensorFlow, PyTorch, Apache Spark, and Federated Learning frameworks like PySyft.

4. Data Preprocessing Tools: Describe any tools or libraries used for data preprocessing and cleaning, as data from distributed sources may require harmonization.

5. Communication Protocols: Define the communication protocols or frameworks used for exchanging data and model updates between distributed nodes. Examples include REST APIs, gRPC, or custom communication protocols.

2.2 Methods:

1. Data Partitioning: Describe how the distributed data is partitioned or divided among different nodes or locations. Options include random partitioning, geographic partitioning, or domain-specific partitioning.

2. Data Privacy and Security: Explain the methods used to ensure data privacy and security, such as encryption, differential privacy, or secure multi-party computation (SMPC).

3. Federated Learning: If applicable, provide details about the federated learning approach, which allows model training to occur on decentralized data without centralizing it. Explain how model updates are aggregated while preserving data privacy.

4. Model Architecture: Specify the machine learning model architecture you're using for your task, whether it's a neural network, decision tree, or other algorithms.

5. Distributed Training Algorithms: Discuss the training algorithms used for collaborative machine learning, such as gradient descent, asynchronous SGD, or federated averaging.

6. Model Synchronization: Explain how model updates are synchronized across distributed nodes to maintain a consistent global model.

7. Model Evaluation: Describe how you evaluate the performance of the distributed machine learning model, including metrics and validation techniques.

8. Scalability: Discuss methods for ensuring that the collaborative machine learning system can scale with the addition of more data sources or nodes.

9. Distributed Optimization: If necessary, detail any optimization techniques used to improve training efficiency, like gradient compression or quantization.

10. Monitoring and Maintenance: Outline strategies for monitoring the health of the distributed system and performing maintenance tasks as needed.

11. Experimental Setup: Specify any experimental setups, including the number of nodes, data sizes, and configurations, to provide context for your results.

12. Results and Analysis: Present the results of your collaborative machine learning experiments and analyze the performance of the distributed model.

13. Discussion: Discuss the implications of your findings, potential limitations, and future directions for research or application.

3. Basic Knowledge of Federated Learning

3.1 Categories of federated learning.

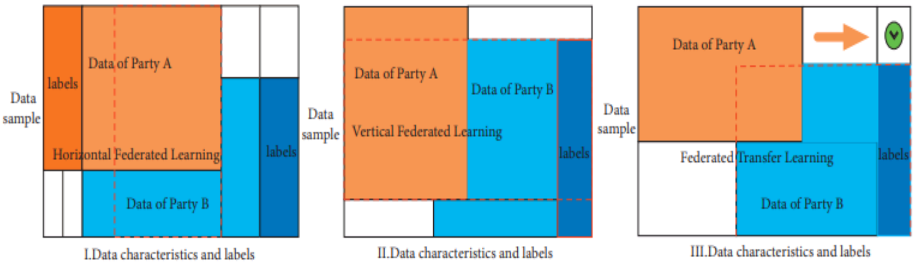
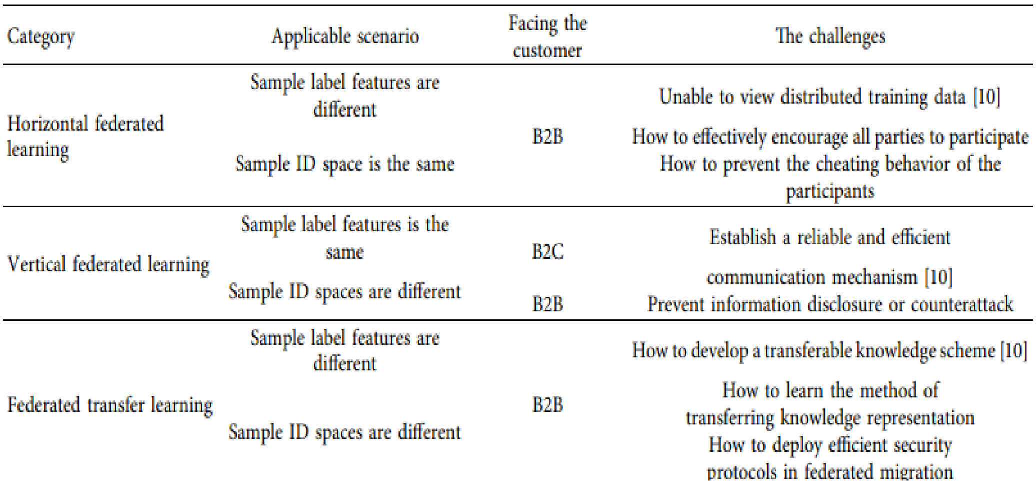

Data sample
labels
Data of Party A
Horizontal Federated Learning
labels
Data of Party B
I.Data characteristics and labels
Data sample
Data of Party A
Data of Party B
Vertical Federated Learning
labels
II.Data characteristics and labels
Data sample
Data of Party A
Federated Transfer Learning
labels
Data of Party B
III.Data characteristics and labels

Figure 1: Three categories of federated learning.

3.2 Comparison of three kinds of federated learning.

Table 1: Comparison of three kinds of federated learning.



Category	Applicable scenario	Facing the customer	The challenges
Horizontal federated learning	Sample label features are different Sample ID space is the same	B2B	Unable to view distributed training data [10] How to effectively encourage all parties to participate How to prevent the cheating behavior of the participants
Vertical federated learning	Sample label features is the same Sample ID spaces are different	B2C B2B	Establish a reliable and efficient communication mechanism [10] Prevent information disclosure or counterattack
Federated transfer learning	Sample label features are different Sample ID spaces are different	B2B	How to develop a transferable knowledge scheme [10] How to learn the method of transferring knowledge representation How to deploy efficient security protocols in federated migration

3.3. Machine Learning.

With the rapid-fire expansion of ML, its models are growing more and more convoluted and efficient(7, 8). Th e core idea of ML is that the computer learns the mapping between input and affair according to being data cross sections fw x ⟶ y, where x is the input, y is the affair, f is the corresponding rule, andw is the parameter to be learned. According to the corresponding relationship, the model predicts the produce value of the coming input. * e purpose of ML is to make the gap between the prognosticated value and the real valuation as small as realizable. * e mathematics is vented as

$$\arg\min_{w} L(x, y, w) = \left\| f_{w}(x) - y \right\|.$$

In traditional ML, such as back propagation neural network (BPNN) and convolution neural networks (CNNs), the learning expansion of this parameter is all concentrated on one central processing unit, and the frequently used methods are gradient descent and a series of improved algorithms. *e core algorithm of FL is very comparable to the Stochastic Gradient Descent (SGD) method [7]. In SGD, a illustration is randomly selected from all samples to engage yourself in the maneuver at each iteration.

3.4. Distributed Machine Learning.

Distributed machine learning is commonly used in scenarios like training deep neural networks on large datasets, processing big data, and handling real-time or streaming data. It allows organizations to harness the power of distributed computing infrastructure to train more complex models and make more accurate predictions.

Distributed Machine Learning Complexity

ALGORITHM 1: Distributed machine learning algorithm.

Server side

(1) Input: data sample X, Y, initial model parameters w0, iterative step μ;

(2) Divide X Perry Y into collections in units of records{ X1,X2, . . . ,Xm}, {Y1, Y1, . . . , Ym}. m indicates the number of clients;

(3) Send Xk, Yk to the client k;

(4) Execute t times (t ≥ 1) for each iteration: send wt−1 to the client;

(5) Receive gradient update gk t from client k; execute wt←wt−1 − μ {m i}1 gk t ;

(6) To determine whether the termination condition is met: if so, it will be terminated; otherwise, it will be executed t←t + 1;

Client side k

(7) Input: Xk, Yk,wt−1;

(8) Batches are randomly selected from Xk,Yk records as training data Xk b,Yk b;

(9) Calculated gradient gk t ←∇L(Xk b, Yk b,wt−1);

(10) Send gk t to server side.

3.5. Federated Learning.

Federated Learning is applied in various domains, including healthcare (e.g., patient data analysis), personalized recommendation systems, and predictive maintenance for IoT devices, where data privacy and efficiency are critical considerations.

$$\arg\min_{w} L(x, y, w) = \sum_{k} p_k L_k(x, y, w),$$

where k is the number of guests, pk is the weight value of the kth customer, and the script for FL is the decentralized multiuser{ F1, F2,..., Fk}. Each customer stoner has the current stoner's data set{ D1, D2,..., Dk}. In deep literacy, these data are sorted out into a data set D{ U1 ∪ U2 ∪ • • • ∪ Uk. * e practice of FL is no longer to simply aggregate them to form a new data set to complete the coming stage of training tasks. Suppose the global model after the completion of a civil modeling task is MFED and the matching training model after aggregation is MSUM. Generally speaking, the global model MFED is performing due to the operation of parameter exchange and aggregation. * ere will be a loss of delicacy during the entire training process; that is, the performance of the global model MFED isn't as good as the performance of the aggregate model MSUM. To quantify this difference, we define the performance of the global model MFED on the test set as VFED, and the performance of the aggregate model MSUM on the test set as VSUM. At this time, the δ- loss delicacy(10) of the model is defined as

$$\left|V_{\mathrm{FED}} - V_{\mathrm{SUM}}\right| < \delta,$$

Where δ is a nonnegative number. However, in actual situations, the aggregation model MSUM cannot be obtained in the end, because the basic requirement of FL is privacy protection. According to Professor Yang's book "Federated Learning" [10], the federated

Federated average algorithm Complexity

ALGORITHM 2: Federated average algorithm

```
(1)  Execute in the coordinator:
(2)  Initialize w0 and broadcast the original model parameter w0 to all participants;
(3)  For each global model update round t { 1, 2, ...,} do;
(4)  *e coordinator determines Ct, that is, determines the set of max(kp, 1) randomly selected participants;
(5)  For each participant k ∈ Ct do in parallel;
(6)  Update the model parameters locally: w(k) t+1 ← participants update (k, wt) (see line 13);
(7)  Send the updated model parameter w(k) t+1 to the coordinator;
(8)  end for
(9)  The coordinator aggregates the received model parameters, that is, using a weighted average for the received model parameters: wt+1$\sum_{k=0}^{n}$ k k1(nk/n)w(k) t+1
(10) The coordinator checks whether the model parameters have converged. If it converges, the coordinator sends a signal to all participants to suspend model training;
(11) The coordinator broadcasts the aggregated model parameter wt+1 parameter-to all participants;
(12) end for
(13) Update in the participant (k, wt)(participants k, ∀k {1, 2, . . . , K} are executed in parallel)
(14) Get the latest model parameters from the server, that is, set w(k) 1,1  wt;
(15) For each local iteration from 1 to the number of iterations S i do;
(16) Batches ← randomly divide the data set Dk into the size of the batch M;
(17) Obtain the local model parameters from the previous iteration, set w(k) 1,i  w(k) B,i−1;
(18) For batch number b do from 1 to batch quantity B { nk/M;
(19) Calculate batch gradient g(b) k ;
(20) Update model parameters locally: w(k) b+1,i←w(k) b,i − ηg(b) k ;
(21) end for
(22) end for
(23) Get the local model parameter update w(k) t+1 { w(k) B,S , and send it to the coordinator (for participants of k ∈ Ct).
```

3.6. Federated Learning Classification

Applications of federated classification include personalized recommendation systems, medical diagnosis with patient data distributed across healthcare institutions, and fraud detection in financial transactions, among others.

3.6.1. Horizontal Federated Learning

Horizontal Federated Learning is a specific approach within the field of Federated Learning that involves training machine learning models across multiple decentralized edge devices or local servers where each device holds different but similar types of data. In Horizontal Federated Learning, the data distribution among participants is typically horizontal, meaning each participant has examples from the same features or attributes but different instances or data points. This approach is well-suited for scenarios where different data sources have a common set of features but their individual data points vary.

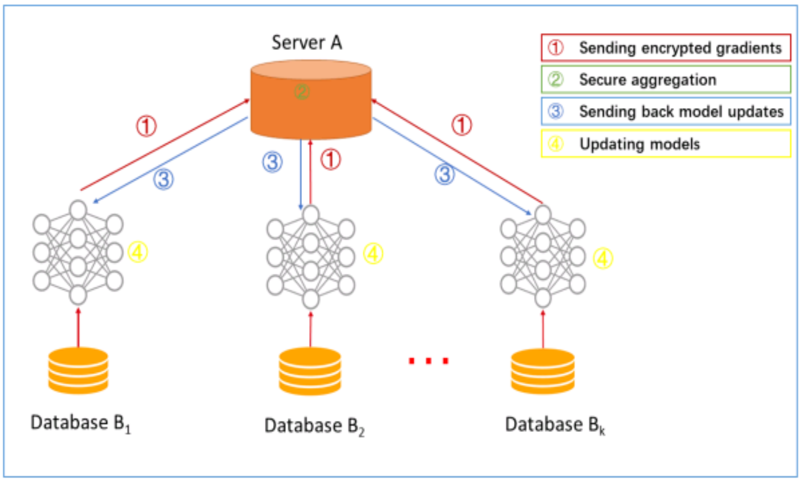


Figure 2. Architecture for a horizontal federated-learning system.

3.6.2. Vertical Federated Learning

Vertical Federated Learning is a specific approach within the field of Federated Learning that involves training machine learning models across multiple decentralized edge devices or local servers where each participant holds different sets of features (attributes) for the same data instances (data points). In contrast to Horizontal Federated Learning, where participants have the same features but different data instances, Vertical Federated Learning is well-suited for scenarios where different data sources have data on the same set of individuals or entities but collect different types of attributes or features about them.

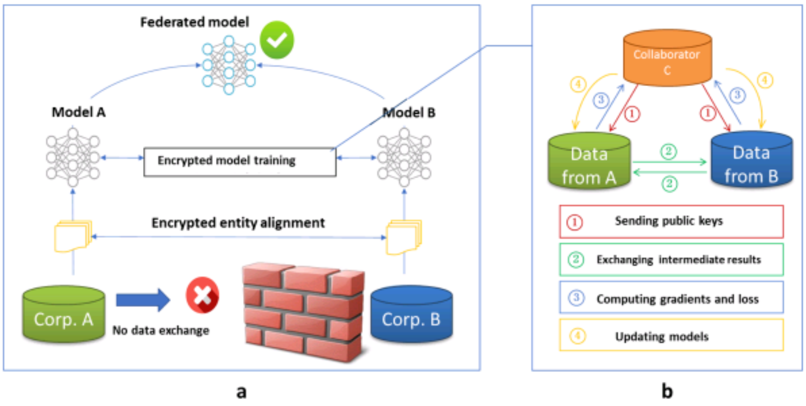


Figure. 3. Architecture for a vertical federated-learning system.

3.6.3. Federated Transfer Learning

Federated Transfer Learning is an extension of the Federated Learning paradigm that combines elements of Federated Learning and Transfer Learning. It's designed for scenarios where multiple decentralized edge devices or local servers, each with their own data, can collaborate to improve a shared model, and there's a need to transfer knowledge from a pre-trained model to this shared model.

3.7 Federated Learning Algorithm Based on Machine Learning

The selection of specific machine learning algorithms for Federated Learning depends on the task, the computational resources available, and privacy requirements. Deep learning models, such as convolutional neural networks (CNNs) for image tasks or recurrent neural networks (RNNs) for sequential data, are commonly used when dealing with complex data types. Additionally, techniques like differential privacy can be incorporated to enhance privacy protection in Federated Learning.

Popular machine learning frameworks and libraries, such as TensorFlow Federated (TFF) and PySyft, provide tools and APIs for implementing Federated Learning algorithms and protocols. These frameworks simplify the development of Federated Learning applications.

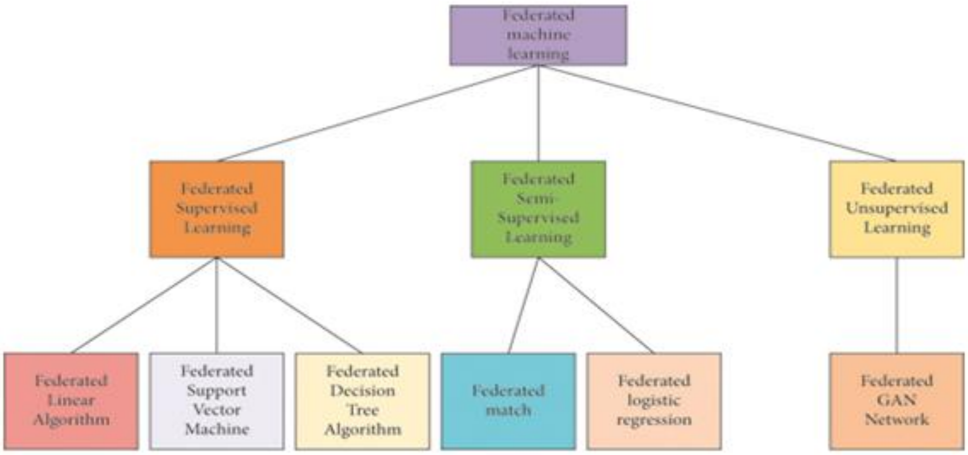

Federated machine learning
Federated Supervised Learning
Federated Semi-Supervised Learning
Federated Unsupervised Learning
Federated Linear Algorithm
Federated Support Vector Machine
Federated Decision Tree Algorithm
Federated match
Federated logistic regression
Federated GAN Network

Figure 4 Federated machine learning classification.

3.7.1 Federated Supervised Learning

Federated Supervised Learning is a specific application of Federated Learning where the goal is to collaboratively train a supervised machine learning model across multiple decentralized edge devices or local servers, while preserving data privacy. In Federated Supervised Learning, each participating device has labeled data, meaning it has input-output pairs for a supervised learning task, and the aim is to collectively improve a global model without sharing raw data.

3.7.2. Federated Linear Algorithm

Federated Linear Algorithm, also known as Federated Linear Regression, is a specific application of Federated Learning in which the goal is to collaboratively train a linear regression model across multiple decentralized edge devices or local servers, while keeping the raw data on each device. Linear regression is a machine learning technique used for modeling the relationship between a dependent variable and one or more independent variables.

3.7.3. Federated Support Vector Machine

The method optimizes and protects the parameters by updating blocks of local modules, attributing feature hashing and other ways. The objective function is as follows:where N is the training data, is the parameters of the model, is the loss at the point , is the regular term of the loss function, and is the hyperparameter to control the penalty. The objective function of support vector machine for traditional ML.

3.7.4. Federated Decision Tree Algorithm

Liu et al. [24] proposed a decision tree-oriented vertical federated learning method, a random forest implementation method based on a centralized FL framework, named as Federated Decision Tree (FDT). Its local participants upload the ranking of performance of their model parameters, not model parameters which the original FL constantly uploaded. Thus, it can greatly reduce communication frequency, a large amount of storage, and computing resources consumed by the encryption.

3.6. Federated Semisupervised Learning

Semisupervised learning is a key issue in the field of ML. It can use as much unlabeled data as possible to complete the task [30]. After FL is added to semisupervised learning, on one hand, FL can be used to ensure that sufficient training data are available, and, on the other hand, semisupervised learning can be used to alleviate the problem of the high cost of client-side scattered data labeling.

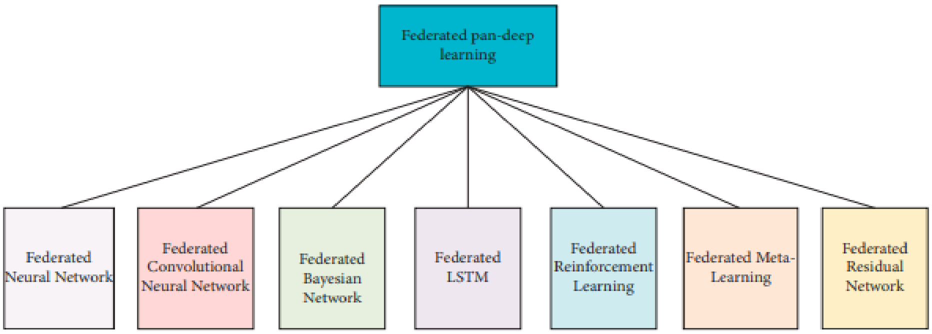

Federated pan-deep learning
Federated Neural Network
Federated Convolutional Neural Network
Federated Bayesian Network
Federated LSTM
Federated Reinforcement Learning
Federated Meta-Learning
Federated Residual Network

FIGURE 3: Federated deep learning classification.

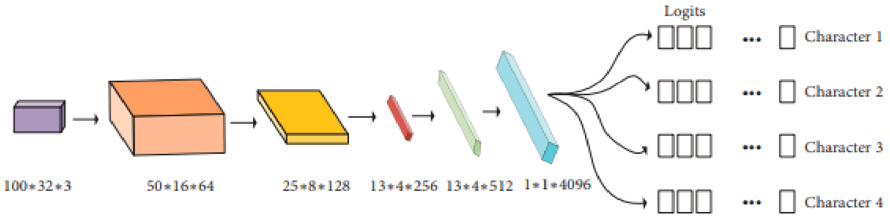


Figure 5. Convolution neural network

3.7 Federated Unsupervised Learning

Federated Unsupervised Learning is an approach within Federated Learning that focuses on collaboratively training unsupervised machine learning models across multiple decentralized edge devices or local servers while preserving data privacy. In unsupervised learning, the goal is to identify patterns, structures, or representations within data without using labeled target values. Here's how Federated Unsupervised Learning works:

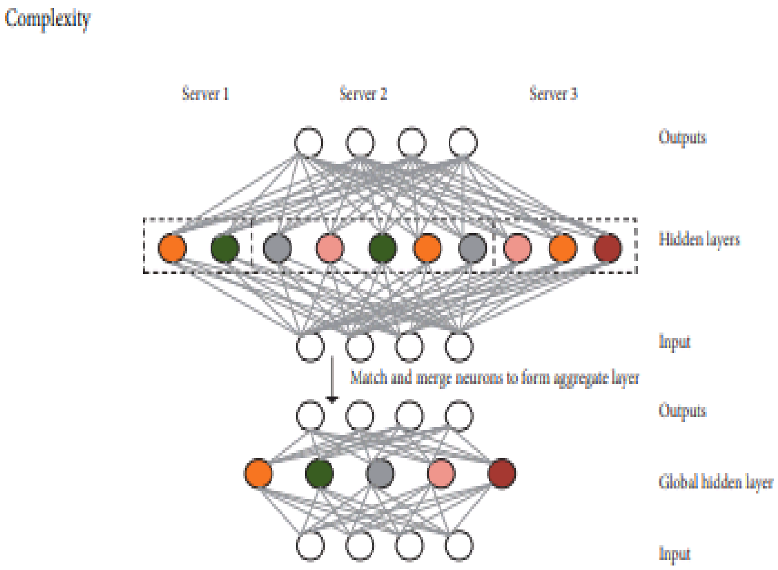


Figure 6. Bayesian Network with hidden layers

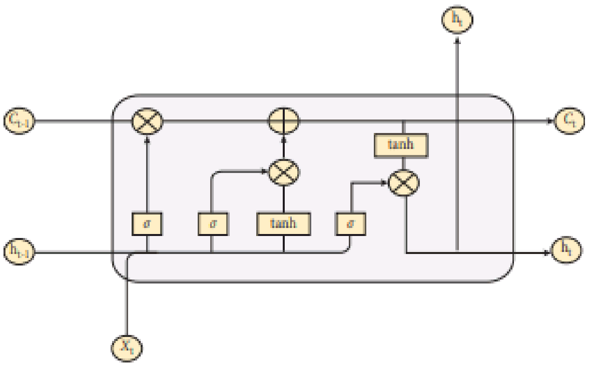


Figure 7. The internal structure of the LSTM network unit

Complexity

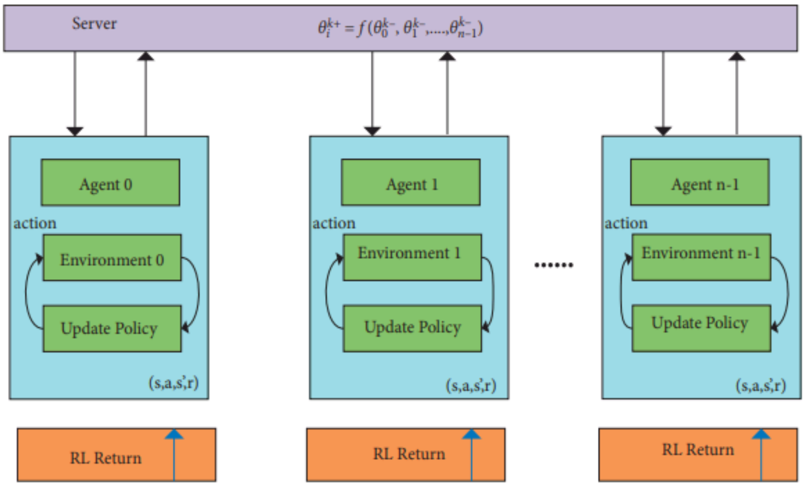


Figure 8. Flowchart of federated reinforcement learning

Complexity

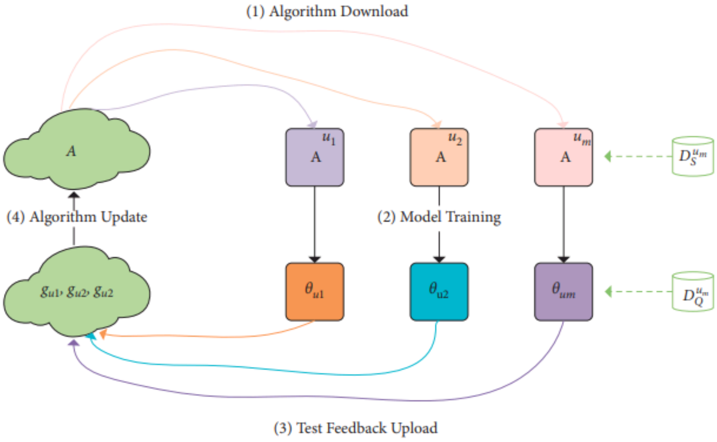


Figure 9. Workflow chart of the federated meta-learning framework

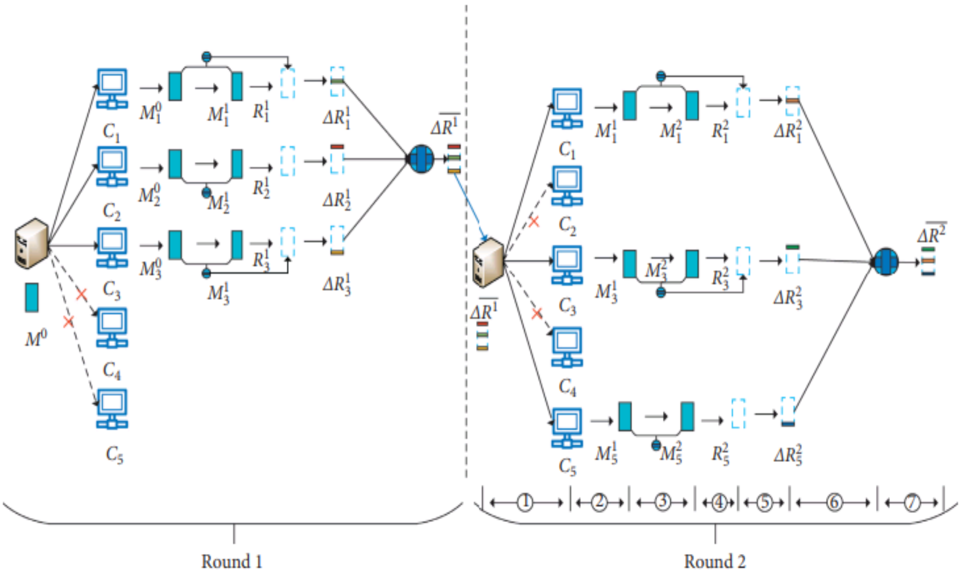


Figure 10. Federated residual network workflow

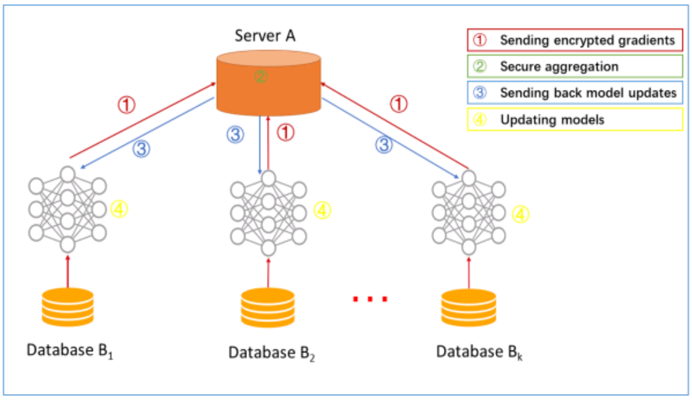


Figure. 10. Architecture for a horizontal federated-learning system.

4. RELATED WORKS

Federated learning enables multiple parties to collaboratively construct a machine- literacy model while keeping their private training data private. As a new technology, allied literacy has several vestments of originality, some of which are embedded on being fields. Below, we explain the relationship between allied literacy and other affiliated generalities from multiple perspectives.

4.1 Sequestration- Conserving Machine Learning

Federated literacy can be considered as sequestration- conserving, decentralized cooperative machine literacy. thus, it's tightly related to multiparty, sequestration- conserving machine literacy. numerous exploration sweats have been devoted to this area in the history. For illustration, the authors of(17, 67) proposed algorithms for secure multiparty decision trees for vertically partitioned data. Vaidya and Clifton propounded self-confident connection mining ground rules(65), secure k- means(66), and a naïve Bayes classifier(64) for vertically partitioned data. The litterateurs of(31) propounded an algorithm for collaboration bylaws on horizontally partitioned data. defend mounting vector motors algorithms have been elaborated for vertically partitioned data(73) and horizontally partitioned data(74). The litterateurs of(16) proposed secure protocols for multiparty direct retrogression and bracket. The authors of(68) proposed secure multiparty grade descent styles. These works all used SMC(25, 72) for sequestration guarantees.

4.2 Federated Learning versus Distributed Machine Learning

Vertical allied literacy at first sight is kindly analogous to distributed machine literacy. Distributed machine literacy covers numerous aspects, including distributed storehouse of training data, distributed operation of calculating tasks, and distributed distribution of model results. A parameter garçon(30) is a typical element in distributed machine literacy. As a tool to accelerate the training process, the parameter garçon stores data on distributed working bumps and allocates data and calculating coffers through a central scheduling knot to train the model more efficiently. For horizontally allied literacy, the working knot represents the data proprietor. It has full autonomy for the original data; it can decide when and how to join the allied literacy. In the parameter garçon, the central knot always takes control; therefore, allied literacy is faced with a more complex literacy terrain. In addition, allied literacy emphasizes the data- sequestration protection of the data proprietor during the model training process. Effective measures to cover data sequestration can more manage with the decreasingly strict data sequestration and data security nonsupervisory terrain in the future.

4.3 Federated Learning versus Edge Computing

Federated literacy can be seen as an operating system for edge computing, as it provides the literacy protocol for collaboration and security. The authors of(69) considered a general class of machine- literacy models that are trained using grade descent – grounded approaches. They dissect the confluence bound of distributed grade descent from a theoretical point of view, grounded on which they propose a control algorithm that determines the stylish trade-off between original update and global parameter aggregation to minimize the loss function under a given resource budget.

4.4 Federated Learning versus Federated Database Systems

Federated database systems(57) are systems that integrate multiple database units and manage the intertwined system as a whole. The allied database conception is proposed to achieve interoperability with multiple independent databases. A allied database system frequently uses distributed storehouse for database units and, in practice, the data in each database unit is miscellaneous. thus, it has numerous parallels with allied literacy in terms of the type and storehouse of data. still, the allied database system doesn't involve any sequestration protection medium in the process of interacting with each other, and all database units are fully visible to the operation system.

5. APPLICATIONS

As an innovative modeling medium that could train a united model on data from multiple parties without compromising sequestration and security of those data, allied literacy has a promising operation in deals, fiscal, and numerous other diligence in which data can not be directly added up for training machine- literacy models owing to factors similar as intellectual property rights, sequestration protection, and data security. Take smart retail as an illustration. Its purpose is to use machine- literacy ways to give guests with substantiated services, substantially including product recommendation and deals services.

The data features involved in the smart retail business substantially include stoner copping

power, stoner particular preference, and product characteristics. In practical operations, these three data features are likely to be scattered among three different departments or enterprises. For illustration, a stoner’s purchasing power can be inferred from the stoner’s bank savings and particular preference can be anatomized from the stoner’s social networks, while the characteristics of products are recorded by ane-shop. In this script, we're facing two problems.

First, for the protection of data sequestration and data security, data walls between banks, social networking spots, ande-shopping spots are delicate to break. As a result, data can not be directly added up to train a model. Second, the data stored by the three parties are generally miscellaneous, and traditional machine- literacy models can not directly work on miscellaneous data.

First, by exploiting the characteristics of allied literacy, we can make a machine- literacy model for the three ACM Deals on Intelligent Systems and Technology,Vol. 10,No. 2, Composition 12. Publication date January 2019. 1214Q. Yang et al.Fig. 5. Data alliance allocates the benefits on a blockchain. parties without exporting the enterprise data, which not only completely protects data sequestration and data security but also provides guests with substantiated and targeted services and thereby achieves collective benefits.

Meanwhile, we can work transfer literacy to address the data diversity problem and break through the limitations of traditional AI ways. thus, allied literacy provides good specialized support for us to make across-enterprise,cross-data, andcross-domain ecosphere for big data and AI. One can use the allied- literacy frame for multiparty database querying without exposing the data. For illustration, suppose that in a finance operation we're interested in detecting multiparty borrowing, which has been a major threat factor in the banking assiduity.

This happens when certain druggies virulently adopt from one bank to pay for the loan at another bank. Multiparty borrowing is a trouble to fiscal stability, as a large number of similar illegal conduct may beget the entire fiscal system to collapse. To find similar druggies without exposing the stoner lists to each other between banksA and B, we can exploit a allied- literacy frame. In particular, we can use the encryption medium of allied literacy and cipher the stoner list at each party and also take the crossroad of the translated list in the confederation. The decryption of the final result gives the list of multiparty borrowers without exposing the other “ good ” druggies to the other party.

Figure. 11. Data alliance allocates the benefits on a blockchain

As we will see below, this operation corresponds to the perpendicular allied literacy frame. Smart healthcare is another sphere that we anticipate will greatly profit from the rising of allied- literacy ways. Medical data similar as complaint symptoms, gene sequences, and medical reports are veritably sensitive and private, yet medical datasets are delicate to collect and live in isolated medical centers and hospitals.

The insufficiency of data sources and the lack of markers have led to an wrong performance of machine- literacy models, which has come the tailback

of current smart healthcare. We image that if all medical institutions are united and partake their data to form a large medical dataset, also the performance of machine- literacy models trained on that large medical dataset would be significantly bettered.

Federated learning combining with transfer literacy is the main way to achieve this vision. Transfer literacy could be applied to fill the missing markers, thereby expanding the scale of the available data and further perfecting the performance of a trained model. thus, allied transfer literacy would play a vital part in the development of smart healthcare and may be suitable to take mortal healthcare to a whole new position.

6. Future Challenges

6.1. Data Privacy Issues.

Under the framework of FL, although the user’s local data do not need to be uploaded to the server, it will be directly used in local modeling. If you do not independently add noise to these local data to protect their security, an attack by a malicious user may take place The attacked federated model and the jointly trained model will lose their balance. In the worst case, the jointly established model cannot be returned to the local client.

6.2. Data Communication Issues.

In the framework of FL, client-side and server-side devices communicate and transmit model parameters or gradients, and its communication rate is more frequent than the traditional distributed machine transmission rate. But each model participating in joint training cannot have the same computing power and stable transmission rate, which will often cause communication instability.

6.3. Data Heterogeneity Issues.

The data of distributed ML are often independent and identically distributed, but FL is different from traditional distributed ML. Devices in FL often exist in the network in a nonindependent and uniformly distributed way. *e data participating in training is generally nonindependent and identically distributed. For example, banks and Internet shopping, although they have the same customers to some extent, their data storage structures are heterogeneous.

6.4. Data Overhead Issues.

In the application scenario of FL, most of the local models that participate in the training need to perform computing and communication tasks on mobile terminal. Because the number of local models involved is very large, it is not only a challenge for the communication but also a great test for computing. FL is not only technical labeling but also a business model.

6.5. Lack of a Trusted Central Server.

In the process of FL, a trusted central server is needed to ensure the privacy and security of users. Some scholars put forward the decentralized algorithm, which is based on the local update scheme of heterogeneous data decentralization training. FL requires a central server to coordinate the training process and receive models uploaded by all clients.

7. Conclusions

This paper discusses the classification and development of FL and several existing problems of FL. It expounds from the point of view of FL algorithm, focusing on federated deep learning on the basis of the introduction of federated ML. In the chapter of federated deep learning, existing deep learning algorithms are discussed from the perspectives of communication, data heterogeneity, privacy protection, and trusted server in FL.

At present, FL is still in the stage of rapid development, and there are still many unsolved problems about ML and deep learning algorithms under the framework of FL. With the further expansion of the amount of data in the future, the implementation of deep learning algorithm is not only a feasible scheme for practicing in the field of artificial intelligence but also a more efficient and comprehensive method for the use of distributed ML and edge data.

In the future, FL will develop incoordination in multiple fields, such as edge computing, blockchain, privacy protection, and other coordinated development to improve the performance of FL and, at the same time, make the commercial value better. Recently, the isolation of data and the emphasis on data privacy became the next challenges for AI, but federated learning has brought us new hope. It could establish a united model for multiple enterprises while local data is protected so that enterprises could work together on data security.

This article generally introduces the basic concept, architecture, and techniques of federated learning, and discusses its potential in various applications. It is expected that, in the near future, federated learning would break the barriers between industries and establish a community where data and knowledge could be shared with safety and the benefits would be fairly distributed according to the contribution of each participant. The bonus of AI would finally be brought to every corner of our lives.

References

[1]. C. Tikkinen-Piri, A. Rohunen, and J. Markkula, “EU general data protection regulation: changes and implications for personal data collecting companies,” Computer Law & Security Report, vol. 34, no. 1, pp. 134–153, 2018.

[2]. Q. Yang, Y. Liu, T. Chen, and Y. Tong, “Federated machine learning: concept and applications,” ACM Transactions on Intelligent Systems and Technology, vol. 10, no. 2, pp. 1–19, 2019.

[3]. J. X. Liu and X. F. Meng, “Survey on privacy preserving machine learning,” Journal of Computer Research and Development, vol. 57, no. 2, pp. 346–362, 2020, in Chinese.

[4]. X. G. Li and F. H. Li, “A survey on differ entail privacy,” Journal of Cyber Security, vol. 3, no. 5, pp. 92–104, 2018, in Chinese.

[5]. H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Ag¨uera y Arcas, “Communication-efficient learning of deep networks from decentralized data,” pp. 1273–1282, 2017, https://arxiv.org/abs/1602.05629.

[6]. J. Konecn´y, H. B. McMahan, F. X. Yu, P. Richt ˇ arik, ´ A. *eertha Suresh, and D. Bacon, “Federated learning: strategies for improving communication efficiency,” 2016, https://arxiv.org/abs/1610.05492.

[7]. G. Yang and Z. S. Wang, “Survey on privacy preservation in federated learning,” Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition), vol. 40, no. 5, pp. 204–221, 2020.

[8]. Z. Chen and B. Liu, “Lifelong machine learning, second edition,” Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 12, no. 3, pp. 1–207, 2018.

[9]. R. Shokri and V. Shmatikov, “Privacy-preserving deep learning,” in Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pp. 1310–1321, Monticello, IL, USA, September 2015.

[10]. Q. Yang and Y. Liu, “Federated learning: the last on kilometer of artificial intelligence,” Journal of Intelligent Systems, vol. 15, no. 1, pp. 183–186, 2020.

[11]. T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, “Federated learning: challenges, methods, and future directions,” IEEE Signal Processing Magazine, vol. 37, no. 3, pp. 50–60, 2020.

[12]. P. Kairouz, H. B. McMahan, B. Avent et al., “Advances and open problems in federated learning,” 2019, https://arxiv.org/ abs/1912.04977.

[13]. H. Kim, J. Park, M. Bennis, and S.-L. Kim, “Blockchained ondevice federated learning,” IEEE Communications Letters, vol. 24, no. 6, pp. 1279–1283, 2019.

[14]. R. Pathak and M. J. Wainwright, “Fed split: an algorithmic framework for fast federated optimization,” 2020, https:// arxiv.org/abs/2005.05238.

[15]. L. Zhu and S. Han, “Deep leakage from gradients,” Lecture Notes in Computer Science-Federated Learning, Springer, Berlin, Germay, pp. 17–31, 2020.

[16]. N. Kilbertus, A. Gascon, M. Kusner, M. Veale, K. Gummadi, ´ and A. Weller, “Blind justice: fairness with encrypted sensitive attributes,” in Proceedings of the 35th International Conference on Machine Learning, pp. 2630–2639, Stockholm, Sweden, July 2018.

[17]. R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, “Cryptonets: applying neural networks to encrypted data with high throughput and accuracy,” in Proceedings of the 33rd International Conference on Machine Learning, pp. 201–210, New York, NY, USA, June 2016.

[18]. C. Dwork, “Differential privacy: a survey of results,” in Proceedings of the International Conference on 8eory and Applications of Models of Computation, pp. 1–19, Springer, Xi’an, China, April 2008.

[19]. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradientbased learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[20]. A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images,” Handbook of Systemic Autoimmune Diseases, vol. 1, no. 4, 2009.