Privacy-Preserving Communication Efficient Approach for Health Data in Distributed Machine Learning

Vaishnavi M[1] Srikanth Vemuru[1,*]

[1]*Department of Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Vaddeshwaram, AP, India-522506*

**Email- vsrikanth@kluniversity.in*

KEYWORDS

Distributed Machine Learning; Privacy Preserving; Communication Efficient; Feature Privacy; Dimensionality Reduction

ABSTRACT

In distributed machine learning systems, communication challenges and privacy concerns arise while transmitting model parameters between nodes. Most of the current solutions have focused on resolving communication-related problems but often fall in effectively safeguarding privacy. Many current distributed machine learning methods primarily concentrate on resolving privacy concerns, sometimes overlooking the vital aspect of safeguarding the privacy of features. These solutions may not sufficiently safeguard the precise attributes of data points used in model training. To address this issue, proposed an Ensemble of Feature Reduction Model (EFRM), which is a pre-processing feature privacy communication method has been implemented. This technique is mainly meant to tackle the concerns related to feature privacy and communication efficiency in distributed machine learning. The goal of this method is to minimize communication inside a node by guaranteeing privacy via data pre-processing to communication between nodes. The experimental findings are evaluated on the Heart Stat log and WDBC datasets using classification metrics such as accuracy and F1 Score. Additionally, the impact on model training and prediction time is addressed.

INTRODUCTION

The Distributed machine learning [1] is machine learning approach where multiple computing resources collectively solve computational problems for efficient model training. DML is adopted in various applications in different industries are getting benefitted by its performance. For example, in healthcare sector, DML has using for patient diagnosis, customized medical care recommendations, and medical image analysis. In finance, for fraud detection and risk assessment to provide personalized financial services. Benefits like efficiency, accuracy and scalability of DML [2-4], there are other obstacles like latency, communication overhead and privacy concerns need to be solved.

The most challenging issue related to latency and communication overhead within distributed machine learning raises problem in training process that will affect the performance of the model whole system where the decision-making should be done quickly. On the other hand, data privacy, when private information is distributed across many nodes becomes complex problem while data transformation and maintaining. To mitigate these challenges, most of the applications are implementing communication optimization strategies like frequent synchronization [5], message compression [6] and feature reduction techniques [7], to reduce

the communication overhead and improve the efficiency of the distributed machine learning system. There are mainly two types of categories in feature reduction methods: filter methods [8] and wrapper methods [9]. Filter methods use scoring methods, such as correlation between the feature and the target variable, to select a subset of input features that are most predictive. Sometimes the domain knowledge is required to filter the subset of the features for predictions will undergoes in manual settings. Wrapper methods, on the other hand, wrap a machine learning model and evaluate the model's performance with different subsets of input features. Among all techniques, Principal components analysis (PCA) [12] is one of the commonly used approaches for dimensionality reduction technique, by eliminating irrelevant or redundant features, will help in streamline the communication process, leading to fast and more efficient model training in distributed machine learning. In article [11], Recursive Feature Elimination (RFE) [10] has its despite of benefits, in the article [11] worked on feature reduction in distributed machine learning have limitations that can impact model performance like loss of information during the mapping of high-dimensional data to low-dimensional space. The identification and acquisition of effective features can be difficult to make dimension reduction one of the most important and difficult tasks in pattern recognition, data mining, and machine learning.

In terms of privacy in Distributed Machine Learning (DML) systems, the involvement of several parties works together without sharing raw data to train a model collaboratively. Although this method has benefits for scalability and privacy protection, it also comes with a set of drawbacks, especially when it comes to protecting the private of sensitive data elements. The possibility of information leaking during the aggregation of model updates is one important problem. As an example, suppose that one participant's data includes very sensitive information like financial transactions or medical histories. Subtle trends in the model updates might unintentionally expose data about the sensitive characteristics, compromising privacy even if the model updates sent during training are encrypted or anonymized. To mitigate above discussed communication and privacy risks and challenges, we proposed a ReArranged pre-processing approach to mitigate the feature privacy and communication issues in distributed machine learning. The main objectives are: first, to ensure the privacy of features by performing pre-processing before transmission across dispersed nodes. To reduce communication costs while training a model on local nodes to improve the model performance.

The structure of the paper is organized as follows. Section 2 is the literature survey taken to develop the proposed method. Section 3 is the proposed method, and the experimental results and analysis are evaluated in section 4. Discussion of proposed model results in section5. Finally, the conclusion of the proposed work is described in section 6.

LITERATURE SURVEY

In distributed machine learning, communication overhead and privacy are the major obstacles need to be addressed. In terms of privacy issues in DML, recent, Federated learning [14], is a kind of distributed machine learning designed to tackle data privacy issues has gained significant interest by improving communication efficiency ability. Prior studies on communication-efficient federated learning have mostly focused on reducing the effects of frequent communication rounds or limits in bandwidth, specifically in the context of Horizontal Federated Learning (HFL) [15]. Several approaches have been investigated to enhance the effectiveness of communication in federated learning [16, 17,18]. These strategies include

selective involvement of clients, minimizing model updates, and using compression methods on models. Client selection is used to optimize communication efficiency by restricting the number of clients involved, which in turn helps to control expenses and minimize the number of parameters that need to be updated in each communication cycle. In the article [19] presented a communication-efficient framework for federated learning (FL) by using a probabilistic device selection strategy. This technique identifies clients with a greater probability of improving the speed of convergence and lowering the loss during model transmission. In similar terms, introduced a similar approach [20] that places restrictions on the quantity of local models sent to the server for aggregation. These limitations are enforced using various techniques, such as random sampling or setting limits depending on the amount of local data or the error rate of local validation.

Minimizing model updates is a possible strategy to reduce the expensive nature of communication process between devices and the central server. The proposed [20] training individual models on devices until they are fully trained, rather than calculating incremental updates. They then used ensemble techniques to accurately capture the model information for each client. This strategy efficiently minimizes the number of communication rounds to just one. The article [21] proposed Partitioned Variational Inference (PVI) as a method for training Bayesian Neural Networks (BNN) in federated learning settings. PVI allows for both synchronous and asynchronous model updates across several computers. When combined with other methods, their approach enables more efficient communication during the training of Binary Neural Networks (BNNs) using non-independent and identically distributed (non-iid) federated data. Additional research on reducing communication overhead in federated learning settings includes the work of [22]. They have developed a one-shot federated learning algorithm called FedKT, which incorporates knowledge transfer techniques. FedKT has been shown to outperform other state-of-the-art federated learning algorithms in terms of communication efficiency within a single round of communication. In their study [23], proposed a federated fusion learning method in which the central server receives distribution parameters of local data instead of model parameters. The central server uses the distribution settings to generate artificial data, which is then included to train a universal machine learning model. This procedure allows the exchange of information between the client and server to occur in a single iteration.

The primary compression techniques used in federated learning primarily focus on compressing gradients, with the goal of reducing training time and transmission costs. However, these tactics often need several cycles of communication. However, our work presents a novel methodology that relies on data compression. This method entails the compression of data saved on separate clients prior to its amalgamation for the ultimate training of the model. By adding safeguards to prevent the publication of any local data, privacy is maintained, and the whole process is consolidated into a single communication cycle.

PROPOSED METHODOLOGY

To achieve the objectives of this article, proposed a ReArrange pre-processing approach to overcome communication-efficiency while protecting feature privacy in distributed machine learning. In ReArrange pre-processing approach, we adopted PCA (Principal Component Analysis) on SFS (Sequential Feature Selection) technique on the original sequence features and Rearranged features.

SEQUENTIAL FEATURE SELECTION

Sequential Feature Selection (SFS) [24] is a machine learning approach used to choose a smaller collection of features from a larger, more complex set of features. The concept is to systematically choose features based on their specific impact on the model's performance, while evaluating the model's performance at all levels.

Let's assume $\boldsymbol{D}$ is the dataset matrix with size of $\boldsymbol{i \times j}$, where $\boldsymbol{i}$ represents rows and $\boldsymbol{j}$ represents number of columns/features. $\boldsymbol{D'}$ is the target column/feature to classify the class which it belongs too. $\boldsymbol{F_k}$ is denoted by the collection of chosen features at the $\boldsymbol{k}^{\text{th}}$ iteration. The evaluation metric score of the model trained using the features in $\boldsymbol{F_k}$ is denoted as Eval($\boldsymbol{F_k}$).

PRINCIPAL COMPONENT ANALYSIS

Principal Component Analysis (PCA) [25] is a commonly used method in data analysis and the reduction of dimensionality. The main objective is to convert the data into a novel coordinate system in a manner that ensures the highest variance, as determined by any projection of the data, is concentrated on the first coordinate (referred to as the first principal component), followed by the second coordinate, and so on.

Suppose a dataset is comprised of $\boldsymbol{i}$ observations of $\boldsymbol{j}$ features. The dataset may be represented as a matrix $\boldsymbol{D}$, where each row corresponds to an observation and each column corresponds to a variable/feature.

Standardization is prior step to use the PCA, which removes the mean ($\overline{\boldsymbol{d_j}}$) from the data and scaling it to unit variance. This step guarantees that every variable makes an equal contribution to the analysis. Where $\boldsymbol{\sigma_j}$ is the standard deviation, $\boldsymbol{d_{ij}}$ is the value of $\boldsymbol{j}$ feature in the $\boldsymbol{i}$ iteration.

$$\boldsymbol{z_{ij} = \frac{d_{ij} - \overline{d_j}}{\sigma_j}}$$

Calculate the standardized data's covariance matrix. The following provides the covariance between variables $\boldsymbol{j}$ and $\boldsymbol{k}$.

$$\boldsymbol{Cov(z_i, z_k) = \frac{1}{n-1}\sum_{i=1}^{n}(z_{ij} - \overline{z_j})(z_{jk} - \overline{z_k})}$$

where $\overline{\boldsymbol{z_j}}$ is the mean of standardized variable $\boldsymbol{j}$. The covariance matrix ($\boldsymbol{C_{m\times m}}$) will result in $\boldsymbol{m \times m}$.

Algorithm 1: Efficient Communication model for feature privacy

---

Host Client: *Local dataset D*
For each host client 1,2, … , *k do*
$\bar{F} \leftarrow$ *sends F*
$\bar{F}$ *will undergoes PCA*

For $\overline{F_{\iota\times J}}$ do

$computes\ z_{ij} \leftarrow \frac{f_{ij} - \overline{f_j}}{\sigma_j}$

$computes\ Cov(z_i, z_k)$
$T \leftarrow [V_1\ V_2 \ldots V_k]$
$MD = D \cdot T$
Train model to on MD

From the $\boldsymbol{C_{m\times m}}$, calculate the eigenvalues ($\boldsymbol{V}$) and eigenvectors ($\boldsymbol{\lambda}$). The directions of maximal variation are represented by eigenvectors, and the quantity of variance along those directions is shown by eigenvalues.

$$\boldsymbol{CV = \lambda V}$$

To create the transformation matrix $\boldsymbol{T}$, sort the eigenvalues[26] in decreasing order and choose the $\boldsymbol{k}$ eigenvectors (principal components) that match the $\boldsymbol{k}$ biggest eigenvalues. Project the original data onto the new feature space, $\boldsymbol{MD = D \cdot T}$, where $\boldsymbol{D}$ is the original dataset and $\boldsymbol{MD}$ is the modified dataset. Combining Principal Component Analysis (PCA) with Sequential Forward Selection (SFS) on the rearranged features in datasets improves data privacy while increasing communication efficiency. SFS minimizes the exposure of sensitive data while optimizing model performance by carefully choosing and adding features one at a time. This iterative process guarantees that only the most relevant and anonymized characteristics are included in the model, in conjunction with PCA's dimensionality reduction capabilities. As a result, this dual technique the model will sent the $\boldsymbol{MD}$ to the server will perfect for data-driven applications where protecting sensitive data is crucial since it improves prediction accuracy while simultaneously fortifying privacy protections.

Table 1 Dataset information

Dataset	No. of Instances	No. of features	No. of classes
Heart stat log [27]	270	14	2
Heart stat log [28]	302	14	2
WDBC [29]	569	32	2

RESULTS

The experimental setup used to evaluate the proposed technique is described in this section. Mainly focused to compare the original sequence of features in a dataset with rearranged features of data, to show the analysis of proposed approach in classification of classes with ensuring feature privacy.

Experiments were performed to evaluate the performance of our proposed method, utilizing various datasets sourced from the UCI repository as detailed in Table 1. Our selection criteria gave precedence to publicly accessible datasets of diverse sizes. We took measures to guarantee a varied selection of dataset sizes, encompassing both small and large components and sample sizes. By employing this methodology, we were able to conduct a thorough evaluation of the performance and resilience of our suggested approach across various data dimensions.

All datasets underwent rigorous pre-processing to eliminate missing values, duplicates, and class imbalances." To address the issue of imbalanced classes within the dataset, oversampling techniques were implemented. To compare the original sequence of features with rearranged features of data to show the analysis of proposed approach in classification of classes with

ensuring feature privacy. The local data is divided into 80% of training data and 20% of testing data. Used the mlxtend.feature_selection and sklearn.decomposition.PCA classes are used for implementation of SFS and PCA. For SFS, 10 fold cross-validations, random state is 42, forward selection is chosen, 'best' keyword is used for k-features to select and scoring=accuracy is taken for the consideration.

ANALYSIS

The results of the experiments from the proposed method are presented in this section. Compared the results on performance metrics such as accuracy and F1-score for classification and for communication-efficient of classifier training and predicting time is calculated for proposed model on original dataset and rearranged features with different classifiers like Logistic regression, Decision Tree, Random Forest, gradient Boosting.

From table 2 and 3 the results of un-shuffled and shuffled features of heart stat log data of 270 samples are gives the insights of the proposed model is ensure the balance between the privacy and utility is achieved. when compared to the real readings, the proposed approach often demonstrates an increase in accuracy across all classifiers. For example, the accuracy of

Table 2 Heart stat log data of 270 samples classification results

Model (270 Un-shuffled)	Actual classifiers readings				Proposed model classifiers readings (EFRM)			
	Accuracy	F1 Score	Training Time (s)	Prediction Time (s)	Accuracy	F1 Score	Training Time (s)	Prediction Time (s)
Decision Tree	0.6852	0.7213	0.0197s	0.0023s	0.7578	0.8235	0.0028s	0.0002s
Random Forest	0.7963	0.8406	0.2958s	0.0128s	0.9074	0.8406	0.1531s	0.0054s
Gradient Boosting	0.7593	0.8116	0.1693s	0.0032s	0.8148	0.9014	0.851s	0.0005s
Logistic Regression	0.8333	0.8732	0.0223s	0.0032s	0.8704	0.8986	0.0039s	0.0002s

Table 3 Heart stat log data of 270 samples of rearranged data classification results

Model (270 shuffled)	Actual classifier readings				Proposed method classifier readings (EFRM)			
	Accuracy	F1 Score	Training time (s)	Prediction time(s)	Accuracy	F1 Score	Training time (s)	Predictive time(s)
Decision Tree	0.7037	0.7333	0.0134	0.0025	0.7778	0.8286	0.0026	0.0002
Random Forest	0.8148	0.8611	0.5196	0.0461	0.8333	0.8696	0.1436	0.0052
Gradient Boosting	0.7593	0.8116	0.5022	0.0031	0.8333	0.8696	0.0790	0.0005
Logistic Regression	0.8333	0.8732	0.179	0.0025	0.8533	0.8832	0.0034	0.0002

the proposed Random Forest model is 0.9074, but the real reading is 0.7963. In the proposed approach, Gradient Boosting and Logistic Regression also show increased accuracy. Although accuracy generally increases, there is variance in the F1 score.

For instance, in the Random Forest classifier, the planned and actual models both have the same F1 score (0.8406). However, the F1 score of the suggested model experiences a

substantial boost when using Gradient Boosting, rising from 0.8116 to 0.9014.The suggested technique often reduces the duration of both training and prediction periods for all classifiers. All models demonstrate a drop in computing time, however, the Random Forest, Gradient Boosting, and Logistic Regression models display particularly significant reductions. Different classifiers exhibit unique responses to the suggested model. The suggested technique improves the accuracy and decreases the prediction time for the Random Forest and Gradient Boosting models, as shown by this example. Nevertheless, it reduces the precision (from 0.6852 to 0.7578) of the Decision Tree model. Even in simpler models, Logistic Regression demonstrates a significant increase in accuracy and F1 score, hence demonstrating the practicality of the proposed technique. Moreover, a significant advantage is that it requires a comparatively shorter timeframe for both training and prediction. We observed that after shuffling the features in dataset with the importance of disease classification the accuracy of the classifiers are extensively improved by table 3, expect LR(Logistic Regression) classifier is maintained same accuracy but after applying the proposed approach has been improved by the accuracy from 0.8333 to 8433. From table 4 and 5 is the subset of heart stat log dataset with 302 instances are taken and with and without shuffling of data is done with different base classifiers but expect logistic regression and gradient boost classifier accuracy remaining classifiers are improved on the shuffling of features in dataset. When compare to training time and predicting time of the classifiers, shuffled features data taken less time with un-shuffled features data when implemented proposed model.

The comparison between actual classifier readings and the classifiers of our proposed model from Table 6 and 7 provides strong proof of the effectiveness of our technique in improving the performance of machine learning models while maintaining computational efficiency on WDBC dataset. Our suggested method effectively improved the accuracy and F1 score metrics for several models applied to shuffled data from the Wdbc dataset. The models include Decision Tree, Random Forest, Gradient Boosting, Logistic Regression, and Ada Boost. By incorporating the proposed method into the Decision Tree model, there is a significant increase in accuracy, rising from 0.9386 to 0.9649. The improvement is matched by a comparable rise in the F1 score, which increases from 0.9176 to 0.9524. Alternative models, such as Random Forest and Gradient Boosting, show similar trends, with the accuracy and F1 score metrics remaining consistent or even improving when employing the suggested technique.

In addition, our approach greatly improves computational efficiency, as seen by the decreased time required for training and prediction in all models. The proposed methodology in Logistic Regression demonstrates significant improvements in accuracy (0.9825) and F1 score (0.9762), along with notable reductions in training and prediction times. In addition, our technique improves the long-lasting and reliable performance stability of AdaBoost models, in both the present and suggested methods. The results provide compelling evidence that our proposed technique significantly enhances the performance of machine learning classifiers, while simultaneously maintaining computing efficiency and ensuring the privacy of features. This highlights its ability to be used in distributed machine learning systems that emphasise the protection of privacy.

Table 4 Heart stat log data of 302 samples classification results

Model (302 Un-shuffled)	Actual classifiers readings				Proposed model classifiers readings (EFRM)			
	Accuracy	F1 Score	Training Time	Prediction Time	Accuracy	F1 Score	Training Time	Prediction Time
Decision Tree	0.6721	0.6774	0.0106	0.0022	0.8033	0.8182	0.0028	0.0002
Random Forest	0.8361	0.8529	0.4383	0.0115	0.8197	0.8358	0.1535	0.0059
Gradient Boosting	0.8361	0.8571	0.1593	0.0028	0.7869	0.8060	0.0866	0.0006
Logistic Regression	0.7049	0.7273	0.0166	0.0016	0.8033	0.8182	0.0041	0.0002

Table 5 Heart stat log data of 302 samples of rearranged data classification results

Model (302 shuffled feature)	Actual classifiers readings				Proposed model classifiers readings (EFRM)			
	Accuracy	F1 Score	Training Time	Prediction Time	Accuracy	F1 Score	Training Time	Prediction Time
Decision Tree	0.7377	0.7333	0.0049	0.0021	0.8033	0.8182	0.0024	0.0001
Random Forest	0.8197	0.8358	0.2651	0.0125	0.8033	0.8182	0.1424	0.0053
Gradient Boosting	0.8361	0.8571	0.1405	0.0024	0.7869	0.8060	0.0779	0.0005
Logistic Regression	0.7049	0.7273	0.0166	0.0016	0.8033	0.8182	0.0032	0.0002

Table 6 WDBC dataset classification results

Model (un-shuffled Wdbc)	Actual classifiers readings				Proposed model classifiers readings (EFRM)			
	Accuracy	F1 Score	Training time (s)	Predictive time(s)	Accuracy	F1 Score	Training time (s)	Predictive time(s)
Decision Tree	0.9386	0.9195	0.0113	0.00026	0.9649	0.9524	0.0019	0.0002
Random Forest	0.9649	0.9524	0.2187	0.0106	0.9649	0.9524	0.2444	0.0134
Gradient Boosting	0.9474	0.9302	0.9474	0.9302	0.9649	0.9524	0.1549	0.0009
Logistic Regression	0.6228	0.0000	0.0125	0.0017	0.9825	0.9762	0.0072	0.0003

Table 7 WDBC rearranged data classification results

Model (shuffled Wdbc)	Actual classifiers readings				Proposed model classifiers readings (EFRM)			
	Accuracy	F1 Score	Training Time	Prediction Time	Accuracy	F1 Score	Training Time	Prediction Time

Decision Tree	0.9386	0.9176	0.0113s	0.0026s	0.9649	0.9524	0.0019s	0.0002s
Random Forest	0.9649	0.9524	0.2187s	0.0106s	0.9649	0.9524	0.2444s	0.0134s
Gradient Boosting	0.9474	0.9302	0.4972s	0.0027s	0.9649	0.9524	0.1549s	0.0009s
Logistic Regression	0.6228	0.0000	0.0125s	0.0017s	0.9825	0.9762	0.0072s	0.0003s

DISCUSSION

In this study, we investigated a privacy preserving model for feature privacy with communication efficient. Our findings indicate the shuffling of features in a dataset to diagnose in healthcare data in most reliable features order can improve the model performance and if the collaboration with the third party can share can with minimum data with ensuing privacy of the data. Observations from the Table 2 and 3, original sequence of feature data and rearranged feature data has improved by applying the proposed approach. Specifically, specific results on Wdbc dataset logistic regression classifiers gave 0.0000 f1-score on the original order of feature and shuffled features but we observed when we applied the proposed approach gave the 0.9647 f1-score and extensively improved the accuracy from 0.6228 to 0.9737.

CONCLUSIONS

In conclusion, the analysis of machine learning models using shuffled and not-shuffled feature datasets, utilizing actual classifier readings and a proposed technique, provided significant information on the effectiveness of privacy-preserving methods. The results demonstrate constant or slightly enhanced model performance using the proposed strategy, notably noticeable in Decision Tree, Random Forest, Gradient Boosting, and Logistic Regression models. The proposed approach shows significant enhancements in computational efficiency, as seen by considerable reductions in training and prediction times across different models and datasets. The increase in efficiency highlights the flexibility and strength of the suggested method, indicating its potential for broad use in machine learning systems that protect privacy. Although AdaBoost demonstrates steady performance and Logistic Regression regularly achieves high accuracy and F1 score metrics, the suggested strategy consistently improves computing efficiency across various feature distributions.

Declarations

Funding: This research received no external funding

Data Availability Statement: The datasets we are used for this study is publicly available https://archive.ics.uci.edu/dataset/145/statlog+heart and https://www.openml.org/search?type=data&sort=runs&id=1510&status=active).

REFERENCES

[1] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konecny, S. Mazzocchi, H.B. McMahan, et al., Towards federated learning at scale: System design, *arXiv* 2019, arXiv preprint arXiv:1902.01046.

[2] Beznosikov, Aleksandr, Martin Takác, and Alexander Gasnikov. Similarity, compression and local steps: three pillars of efficient communications for distributed variational inequalities. *Advances in Neural Information Processing Systems* 2024.

[3] Opz, Florence, and Fawgan Gen. Decoupling Strategies for Federated Machine Learning with Database as a Service.2024.

[4] Llasag Rosero, Raúl, et al. Label synchronization for Hybrid Federated Learning in manufacturing and predictive maintenance. *Journal of Intelligent Manufacturing* 2024.

[5] He, Yutong, Xinmeng Huang, and Kun Yuan. Unbiased Compression Saves Communication in Distributed Optimization: When and How Much?. *Advances in Neural Information Processing Systems* 2024.
[6] Liu, Jia, et al. A feature selection method based on multiple feature subsets extraction and result fusion for improving classification performance. *Applied Soft Computing* 2024.
[7] Qian, Feng, et al. Unsupervised Intense VSP Coupling Noise Suppression with Iterative Robust Deep Learning. *IEEE Transactions on Geoscience and Remote Sensing* 2024.
[8] Hammoud, Ahmad, et al. Coronary Heart Disease Prediction: A Comparative Study of Machine Learning Algorithms. *Journal of Advances in Information Technology* 2024.
[9] Magboo, Vincent Peter C., and Ma Sheila A. Magboo. Cardiovascular disease prediction with imputation techniques and recursive feature elimination. *AIP Conference Proceeding*s 2023.
[10] B. Zhang, J. Geng, W. Xu and L. Lai, Communication efficient distributed learning with feature partitioned data, *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, USA, 2018, pp. 1-6, doi: 10.1109/CISS.2018.8362294.
[11] Petscharnig, Stefan, Mathias Lux, and Savvas Chatzichristofis. Dimensionality reduction for image features using deep learning and autoencoders. *Proceedings of the 15th international workshop on content-based multimedia indexing*. 2017.
[12] Liu, Ji, et al. From distributed machine learning to federated learning: A survey. *Knowledge and Information Systems* 2022.
[13] Asad, Muhammad, Ahmed Moustafa, and Takayuki Ito. Fedopt: Towards communication efficiency and privacy preservation in federated learning. *Applied Sciences* 2020.
[14] Liu, Tong, et al. LEARN: Selecting Samples Without Training Verification for Communication-Efficient Vertical Federated Learning. GLOBECOM 2023-2023 *IEEE Global Communications Conference. IEEE*, 2023.
[15] Sun, Jingwei, et al. Communication-efficient vertical federated learning with limited overlapping samples. *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023.
[16] Khan, Afsana, Marijn ten Thij, and Anna Wilbik. Communication-efficient vertical federated learning. *Algorithms* 2022.
[17] Chatterjee, C.; Roychowdhury, V.P.; Chong, E.K. On relative convergence properties of principal component analysis algorithms. *IEEE Trans. Neural Netw*. 1998.
[18] Chen, M.; Shlezinger, N.; Poor, H.V.; Eldar, Y.C.; Cui, S. Communication-efficient federated learning. *Proc. Natl. Acad. Sci. USA* 2021. [CrossRef]
[19] Guha, N.; Talwalkar, A.; Smith, V. One-shot federated learning. *arXiv* 2019.
[20] Bui, T.D.; Nguyen, C.V.; Swaroop, S.; Turner, R.E. Partitioned variational inference: A unified framework encompassing federated and continual learning. *arXiv* 2018, arXiv:1811.11206.
[21] Li, Q.; He, B.; Song, D. Practical one-shot federated learning for cross-silo setting. *arXiv* 2020, arXiv:2010.01017.
[22] Kasturi, A.; Ellore, A.R.; Hota, C. Fusion learning: A one shot federated learning. *In Proceedings of the International Conference on Computational Science, Krakow, Poland, 16–18 June* 2020.
[23] Rückstieß, Thomas, Christian Osendorfer, and Patrick Van Der Smagt. "Sequential feature selection for classification." *AI 2011: Advances in Artificial Intelligence: 24th Australasian Joint Conference, Perth, Australia, December 5-8,* 2011. Proceedings 24. Springer Berlin Heidelberg, 2011.
[24] Abdi, H.; Williams, L.J. Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.* 2010. [CrossRef]
[25] Fox, R.; Kapoor, M. Rates of change of eigenvalues and eigenvectors. *AIAA J.* 1968.

[26] Statlog (Heart). UCI Machine Learning Repository. https://doi.org/10.24432/C57303.
[27] Detrano, R.; Janosi, A.; Steinbrunn, W.; Pfisterer, M.; Schmid, J.; Sandhu, S.; Guppy, K.; Lee, S.; Froelicher, V. International application of a new probability algorithm for the diagnosis of coronary artery disease. Am. J. Cardiol. 1989.[CrossRef]
[28] Wolberg,William, Mangasarian,Olvi, Street,Nick, and Street,W. Breast Cancer Wisconsin (Diagnostic) 1995. UCI Machine Learning Repository. https://doi.org/10.24432/C5DW2B