Federated Learning on Transcriptomic Data: Model Quality and Performance Trade-Offs

Anika Hannemann[1,2], Jan Ewald[2], Leo Seeger[2], and Erik Buchmann[1,2]

[1] Dept. of Computer Science, Leipzig University
[2] Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI) Dresden/Leipzig, Leipzig University, Germany surname@cs.uni-leipzig.de

Abstract. Machine learning on large-scale genomic or transcriptomic data is important for many novel health applications. For example, precision medicine tailors medical treatments to patients on the basis of individual biomarkers, cellular and molecular states, etc. However, the data required is sensitive, voluminous, heterogeneous, and typically distributed across locations where dedicated machine learning hardware is not available. Due to privacy and regulatory reasons, it is also problematic to aggregate all data at a trusted third party. Federated learning is a promising solution to this dilemma, because it enables decentralized, collaborative machine learning without exchanging raw data.
In this paper, we perform comparative experiments with the federated learning frameworks TensorFlow Federated and Flower. Our test case is the training of disease prognosis and cell type classification models. We train the models with distributed transcriptomic data, considering both data heterogeneity and architectural heterogeneity. We measure model quality, robustness against privacy-enhancing noise, computational performance and resource overhead. Each of the federated learning frameworks has different strengths. However, our experiments confirm that both frameworks can readily build models on transcriptomic data, without transferring personal raw data to a third party with abundant computational resources.

Keywords: Federated Learning · Cell Type Classification · Disease Prognosis

1 Introduction

Machine learning has the potential for a paradigm shift in healthcare, towards medical treatments based on individual patient characteristics [32,9]. For example, precision medicine uses biomarkers, genome, cellular and molecular data, and considers the environment and the lifestyle of patients [17,19]. Machine learning on large scale genomic and transcriptomic data enables the identification of disease subtypes, prediction of disease progression and selection of targeted therapies. Therefore, models need to be trained on large, diverse patient cohorts (sample size) with high-resolution genetic characterization (number of features). This is challenging: The data is commonly distributed across multiple

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

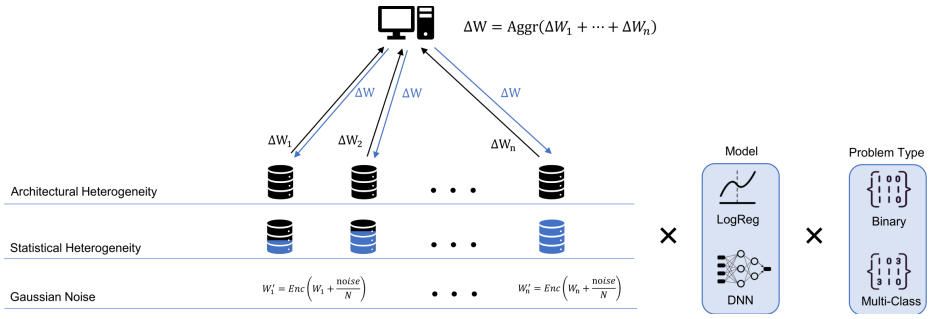

Fig. 1: Key challenges of Federated Learning.

healthcare institutions that may not possess the high-performance computing resources needed to build large deep-learning models. The sensitive nature of genomic and transcriptomic data presents privacy challenges. Genomic mutations and markers could even allow to re-identify individuals and their relatives [27]. This disallows to freely share such data with centralized aggregators.

Federated learning (FL), as shown in Figure 1, allows for decentralized model training across multiple data sets without requiring to transfer sensitive raw data [24,29]. Only model parameters are exchanged between the aggregator and participating clients, and the overall computational burden is effectively shared. Adding noise [3,5,8] to shared parameters could further increase privacy.

In this paper, we investigate the technical and conceptual challenges that arise when implementing FL on transcriptomic data. Figure 1 illustrates our four key challenges: *Architectural Heterogeneity* refers to different numbers of clients with varying computational capabilities. *Statistical Heterogeneity* relates to data distributions and sizes. *Gaussian Noise* addresses the impact of applying noise to the data, e.g., to achieve Differential Privacy, with different models (Logistic Regression and Sequential Deep Learning) and problem types (Binary and Multi Label). Finally, *Resource Consumption* addresses storage, communication overhead and training times. To explore these challenges, we have conducted comparative experiments with two state-of-the-art FL frameworks – TensorFlow Federated (TFF) and Flower (FLWR) – and transcriptomic data. In particular, we make the following contributions:

- We train disease prognosis and cell type classification models with TFF and FLWR using hyperparameter tuning, and we measure the model quality.
- We analyze the effects of the number of clients, the amount of training data and data distribution on the global model quality.
- We measure the impact of Gaussian noise, locally applied to the weights, on the global model quality.
- We compare memory consumption, run-times and network traffic of TFF and FLWR from both the client's and the aggregator's perspective.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

We have demonstrated that FL frameworks can be readily applied to precision medicine applications. Even more, we obtained an excellent global model with an AUC of up to 0.98 for disease prognosis and cell type classification with transcriptomic data. This performance is robust in the presence of diminishing data quality, increasing clients and diverse data distributions, and it reduces the necessary computational resources for the individual medical institution.

Paper structure: Section 2 introduces related work. Section 3 describes our methodology. Section 4 presents our experimental results. Section 5 concludes.

2 Related Work

This section reviews related work in the fields of FL and its applications in precision medicine, and discusses technical challenges and FL frameworks.

Federated Learning and Applications FL [24] allows distributed clients $\mathcal{N}_1, ..., \mathcal{N}_n$ to collaboratively train a model $\mathcal{M}_{glob}$ without sharing raw data. Instead of centralizing the data sets like in traditional machine learning, each client $\mathcal{N}_i$ trains a local model $\mathcal{M}_i$ on its own data set. The model parameters are then shared with a central aggregator, which aggregates them to create a global model $\mathcal{M}_{glob}$. This process is repeated as more data is collected, with clients continuously updating their local models and forwarding the updated parameters to the aggregator. Raw data remains with the respective clients and is never transmitted.

FL is used for collaboration across medical institutions, hospitals, health care insurers or other entities [34,4,10,20]. Some problems relevant to medicine were addressed by [32] and [9] who trained a federated model to model Alzheimer's and Parkinson's disease. Beguier *et al.* [5] present a differentially private and federated cancer occurrence prediction based on genomic data. For practical implications and benchmarking of frameworks for FL, however, there is less literature available. The multi-class data set we use in experiments [16], to our knowledge, has not been modeled by any FL system. The authors of the binary data set propose a collaborative learning method named swarm learning without an aggregator [30]. This method achieves very good model quality, but does not take into account many challenges that arise when bringing FL in to production. We identified four major challenges with heterogeneity in data distribution, participating clients, consumption of computational resources and privacy:

Technical and Conceptual Challenges The issue of **statistical heterogeneity** of training data in FL encompasses both the distribution and size of the data. This challenge involves dealing with the non-IID (non-independent and identically distributed) nature of distributed data, which can lead to skewed or biased model training [26,18]. Additionally, the size of data sets of each client can vary significantly, where smaller data sets may not adequately represent the population, impacting model quality and generalizability. Fu *et al.*[12] showed that not only data heterogeneity can influence the model's quality, but also the varying number of clients which we call **architectural heterogeneity**. Navigating

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

these aspects of statistical heterogeneity is crucial for ensuring the robustness and efficacy of FL models. In the context of transcriptomic data and health-care institutions both issues are very common since hospitals vary greatly in their sizes and specialization or different laboratories introduce bias due to small differences in sequencing protocols and machinery [29].

Furthermore, while FL is designed to enhance **privacy** by training models locally and sharing only model updates, ensuring the privacy and security of these updates against potential inference attacks remains a critical concern. Multiple works showed, that there is no formal privacy guarantee for FL without additional privacy-enhancing techniques [18]. Recent publications [7,33] showed that baseline FL is vulnerable to reconstruction attacks, while others [13,25] successfully performed Membership Inference Attacks (MIA). Multiple works [8,5,3] explored Differential Privacy in a FL scenario for medical data. Differential Privacy protects the privacy of individual data points in a training data set while allowing ML models to benefit from the overall information. It adds controlled noise to data or model parameters, making it difficult to infer specifics about individual entries. However, the application of noise usually comes with information loss. For transcriptomic data in combination with clinical patient data, this is a dilemma since biomarker signals are often weak for multi-factorial diseases. Hence, privacy levels need to be carefully chosen to find a trade-off between model quality, which is highly critical in medical applications, and privacy.

Finally, **resource consumption** presents another challenge for FL. The diverse and potentially resource-limited nature of participating clients in a FL network can lead to inefficiency and delay [18,12]. Furthermore, the communication required for model updates and synchronization in FL adds to network bandwidth demands, which can be a bottleneck in resource-limited environments.

FL Frameworks The field of federated learning is rapdily evolving, and there are many existing open-source FL frameworks, such as TensorFlow Federated (TFF) [2,14], Flower (FLWR) [6], PySyft [35], FATE [22] and FedML [15]. These frameworks vary in terms of their features, ease of use, and specific use cases. The choice of a federated learning framework typically depends on the specific requirements and constraints of the application.

Both PySyft and TFF are well established and benefit from a large community support. While TFF is based on the TensorFlow ecosystem, PySyft works primarily with PyTorch. Both PySyft and FATE provide multiple optional privacy-enhancing methods such as Differential Privacy and Secure Multi Party Computation. FLWR is designed to be framework-agnostic and can work with various machine learning frameworks, including TensorFlow, PyTorch, and others. In terms of abstraction level, Flower's API is more high-level and is, therefore, supposed to be more user friendly than TFF and PySyft. While FLWR and FATE only allow simulation and cluster deployment, FedML provides a flexible and generic API and allows on-device training. Also, FedML can be used for various network topologies such as Split Learning, Meta FL and Transfer-Learning.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

3 Methodology

This section explains our experimental concept, the data sets we used, and the architectures of the machine learning models for our experiments.

3.1 Experimental Concept

To quantify the impact of our four key challenges on FL with transcriptomic data, we measure the quality of a global model obtained by FL on centralized data first. With this baseline, we conduct comparative experiments to explore the effects of *Architectural* and *Statistical Heterogeneity* on the model quality. We explore the impact of *Gaussian noise* to enhance privacy and measure its effect on the model. Furthermore, we assess the *Resource Consumption* at the aggregator and the clients.

To draw robust conclusions, we vary problem type and model architecture. In particular, we conduct experiments not only on a binary-labeled data set but also on a multi-class data set. This aligns with clinical research, which typically covers a variety of diseases and research questions rather than a single condition. In multi-class problems, the complexity increases as the model must differentiate between multiple, often overlapping conditions.

We decided to use the FL frameworks **TensorFlow Federated** (TFF) [2,14] and **Flower** (FLWR) [6]. Our choice was driven by multiple factors: We prioritize documentation and usability. Furthermore, we are interested in exploring horizontal FL with frameworks that have programming interfaces at different levels of abstraction. Finally, frameworks with the same communication protocol allow to compare the network performance.

3.2 Data Sets

We use two data sets. The Acute Myeloid Leukemia data set [31] was previously obtained from 105 studies, resulting in 12,029 samples with binary labels. We call it the **binary-labeled data set**. It consists of gene expressions by microarray and RNA-Seq technologies from peripheral blood mononuclear cells (PBMC) of patients with either a healthy condition or acute myeloid leukemia (AML).

The **multi-class data set** includes expression profiles generated by single-cell RNA-Seq for cell types of the human brain, in particular the middle temporal gyrus (MTG). The data set was published by [16], who isolated sample nuclei from eight donors and generated gene expression profiles by single-cell RNA-Seq for a total of 15928 cells (samples) describing 75 distinct cell subtypes. We reduced the number of classes (cell types) to make the data set more suitable to experiment with class imbalance. For that we selected only the the five most abundant cell types (classes) leading to 6931 cells (samples) for training. We preprocess both data sets as in previous analyses and benchmarks [31,30,17,16].

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

3.3 Model Architectures

We experiment with a **logistic regression model** and a **deep-learning model**. Following [30], the deep-learning model uses a sequential neural network architecture. It consists of a series of dense layers, each with 256, 512, 128, 64, and 32 units, all activated using the 'relu' activation function. Dropout layers with dropout rates of 0.4 and 0.15 prevent overfitting. The configuration of the output layer is based on the number of classes in the data set.

Table 1: Optimized Hyperparameters

Data Set	**Model**	**Hyperparameters**
Binary	Deep Learning	Adam, L2: 0.005, Epochs: 70
	Logistic Regression	SGD, L2: 0.001, Epochs: 8
Multi Class	Deep Learning	Adam, L2: 0.005, Epochs: 30
	Logistic Regression	SGD, L2: 1.0, Epochs: 10

For our baseline and for hyperparameter tuning, we train both models on centralized data. In particular, the hyperparameter space was randomly searched to find Cross Entropy as optimal loss function with a batch size of 512. Hyperparameters that differ in the respective combinations of model and data are summarized in Table 1. We denote the rounds of training based on the local epochs, so that the total number of epochs remains a constant. Assume one round of training and two clients using 100 local training epochs. With two rounds of training, this would be 50 local epochs for both clients.

4 Experiments

In this section, we describe our experimental setup and our analysis results regarding model quality, data quality and resource consumption.

4.1 Experimental Setup

We perform all experiments using one CPU core from an AMD(R) EPYC(R) 7551P@ 2.0GHz - Turbo 3.0GHz processor and 31 Gigabyte RAM for each client. The network is a 100 Gbit/s Infiniband. We measure the network traffic with tshark [1]. No GPU is used during the experiments. To ensure resource parity among different frameworks and the central model, each training process is bound exclusively to one CPU core. Our experiments are implemented in Python. For preprocessing and data loading, we used the libraries Pandas [23] and Scikit-learn [28]. Both the logistic regression model and the deep-learning model were implemented using Keras, with default settings and federated algorithms from FLWR and TFF [2,14,6]. The default of FLWR is a federated averaging strategy

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

in a client-aggregator setup. Additionally, we compute the average score of each metric for every client. The default building function in TFF uses a robust aggregator without zeroing and clipping of values as model aggregator. The clients of TFF all use the eager executor of TFF and are loading the data with a custom implementation of the data interface of TFF [2]. In this configuration, FLWR and TFF implement the federated averaging algorithm with a learning rate set to 1.0 [24]. The code to our experiments can be found at [21].

For our experiments, we tested combinations of Logistic Regression (LogReg) and Sequential Deep Learning (DL) models together with binary problems (Binary) and multi-label problems (Multi). In the figures, we abbreviate the combinations of models and problem types as Binary LogReg, Multi LogReg, Binary DL and Multi DL. For each combination, we tested 3, 5, 10, 50 clients and 1, 2, 5, 10 rounds of training, and we measured model quality and computational resources used. To analyze the effect under investigation, we iterated over the respective other parameter and reported the averaged result, i.e., when varying the number of clients, we conducted tests for each training round configuration and presented the cumulative effect observed across all training rounds.

4.2 Model Quality

To explore the impact of heterogeneity on the global model, we use a 5-fold cross validation and compute the Area Under the Curve (AUC). A higher AUC value (closer to 1) indicates a better model, that distinguishes between the classes more effectively. We compare the AUC of the centralized baseline with the AUC obtained with FL and varying numbers of clients and training rounds. In the following, we explain our key findings.

Finding 1: Boosting training rounds does not always enhance model quality. We assumed from previous work (cf. Sec. 2) that an increasing number of training rounds improves the quality of the global model. To examine the impact of frequency of weight updates among clients during training, we varied the numbers of training rounds and kept the total number of epochs constant. This approach is consistent with our round configurations optimized with hyperparameter tuning, as described in Section 3.

Consider Figure 2). The left two columns of diagrams show the results of our experiments with varying numbers of rounds, the right ones varying numbers of clients over the respective other variable. The top line of diagrams were obtained with deep learning models, the bottom line with logistic regression. We find that the AUC of the global model does not improve significantly with the number of rounds for logistic regression. Deep learning benefits slightly with more rounds of weight updates (see Figure 2). For non-balanced data sets, matching most real-world FL scenarios, updating rounds prove to be more effective to mitigate class-imbalance across different clients (see Figure 3). Thus, healthcare institutions should carefully chose the number of rounds in FL applications, based on the data distribution and machine learning algorithms used.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

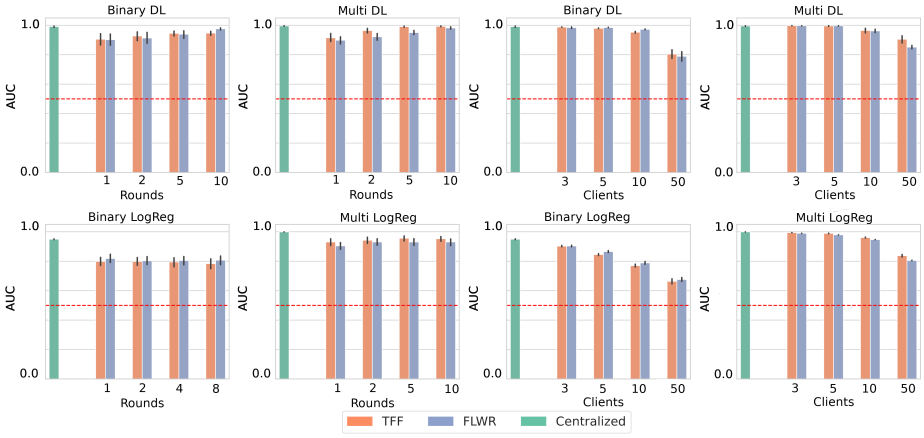

Fig. 2: AUC with respect to an increasing training rounds and clients

Finding 2: Fewer clients and more data increases model quality. To analyze the effect of the number of clients, we split the training data in disjoint subsets and distributed them to clients. Each subset has the same size and class distribution as the whole data set. The data on each client is then split into training and test data with a ratio of 80:20. The test data of all clients is combined and the aggregated FL model is evaluated on this test data. Then, both FL frameworks were used to train a global model.

For a small number of clients, Figure 2 reports that the AUC of FL is similar to our baseline (AUC 0.98), proving that FL can reach centralized model quality in real-world scenarios. A slight decrease for 10 clients can be discovered which is followed by a drop in AUC in the extreme case of 50 clients. This results from the reduced size of local training data: As the number of clients increases, the data set is divided among them, resulting in a reduction in the training data available for local training. The limited data size does indeed reflect a possible scenario where small healthcare institutions want to attend to a FL scenario. We conclude that clients should only allowed when they provide a substantial number of samples on which local model training can be performed.

Finding 3: Model quality is driven by models, not by frameworks. Figure 2 shows that the logistic regression model has an overall lower model quality than the deep learning model for the multi-class problem, as its highest AUC value is 0.90 for both frameworks. In contrast, the deep learning model has an AUC of 0.99 for TFF and 0.98 for FLWR. This emphasizes the superiority of deep learning for this data set, and underlines that the model (and their hyperparameterisation) must fit to the problem.

Benchmarks for a direct comparison of model quality are rare, especially for transcriptomic data. We wanted to learn if there is a difference between FLWR and TFF across the tested scenarios. As Figure 2 illustrates, if all parameters and

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

aggregation algorithms for FLRW and TFF are configured in the same way, both frameworks deliver a similar model quality. This observation holds for various configuration. Healthcare institutions are therefore free to select FL frameworks based on functionality (e.g. privacy-support), usability and computational resource demand, instead of concerning model quality.

Finding 4: Class imbalance impairs federated learning. A challenge for FL is that data points are usually not independent and identically distributed (IID), leading to statistical heterogeneity. With transcriptomic data, a balanced class distribution among all clients seems unlikely. Therefore, we explored the effect of data imbalance on the global model's quality. We investigate IID- and non-IID-data by methodically increasing the imbalance to find a sweet spot of imbalance and model quality.

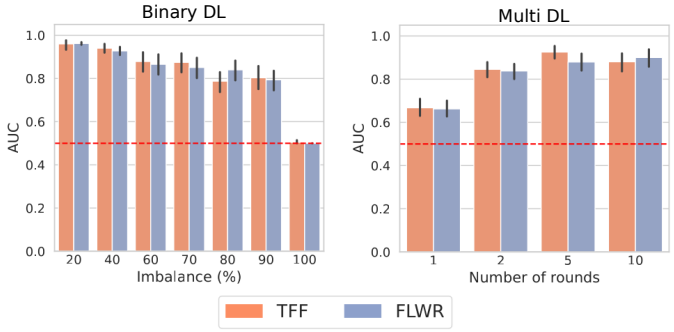

Fig. 3: AUC with respect to an increasing class imbalance and training rounds over all imbalance configurations

We start our experiments with a number of clients equal to the number of classes, and all classes are equally distributed among the clients. Subsequent experiments increase the number of samples from one class while reducing the number of samples from another class. This process is repeated independently for each client and class, until each client contains samples from a single class. Due to lack of space, we present a visualization of only a subset of the conducted experiments (Figure 3). Detailed visualizations and results can be found at [21].

We compare the resulting AUC with an equally-distributed data set. As Figure 3 shows, deep learning can indeed fight class imbalance. However, if the imbalance exceeds a certain threshold, a sudden drop in AUC can be expected. The threshold depends on the data set and machine learning algorithm. Our results indicate that a non-IID distribution not necessarily results in poor model quality. But, the model quality in the presence of non-IID is strongly dependent on the problem type and data set. This can be further increased with the appropriate model selection. As Figure 3 shows, deep learning is robust with a drop in AUC at 90% imbalance. Whereas more training rounds does not improve the robustness for logistic regression, it does for deep learning. We also inves-

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

tigated the effect of multiple rounds to a non-IID setting. Again, we increased the training rounds over all configurations in data distributions and report the average. The increase in a number of training rounds leads to an improvement of model quality for deep learning with non-IID data (see Figure 3), but does not affect the quality of logistic regression, regardless of the problem type [21]. Thus, healthcare institutions need to consider that the model quality depends on minimal number of samples per class. This number depends on aspects like the training algorithm.

4.3 Data Quality and Privacy

There are many anonymization approaches such as Differential Privacy, which apply noise to the data. Because TFF and FLWR have different levels of support for Differential Privacy, we have chosen a general approach to investigate the impact of anonymization on model quality: We add Gaussian noise to the local parameters before aggregation.

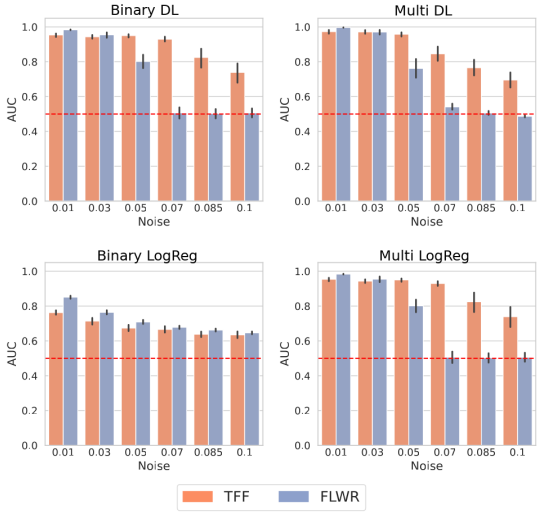

Fig. 4: AUC with respect to an increasing local Gaussian noise

Finding 5: Adding noise to the weights has a strong impact on model quality. Our experiments use five clients and varying noise parameters 0.01, 0.03, 0.05, 0.07, 0.085, 0.1. We observe that all model and problem types deal with some noise. However, at some point AUC drops to approx. 0.07 – 0.085. TFF copes with noise better than FLWR, possibly because TFF uses regularization techniques that mitigate the impact of noisy updates. Increasing the training rounds does not improve the model quality, but slightly decreases AUC. Thus, if healthcare institutions want to apply differential privacy, sophisticated approaches are required to ensure model quality.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

4.4 Computational Resources

When analyzing the computational resources of a federated system, both the local and the global perspective are relevant. We investigate the aggregated training time, memory consumption and network traffic for the clients and the aggregator. Again, we present our main findings and refer to [21] for detailed results.

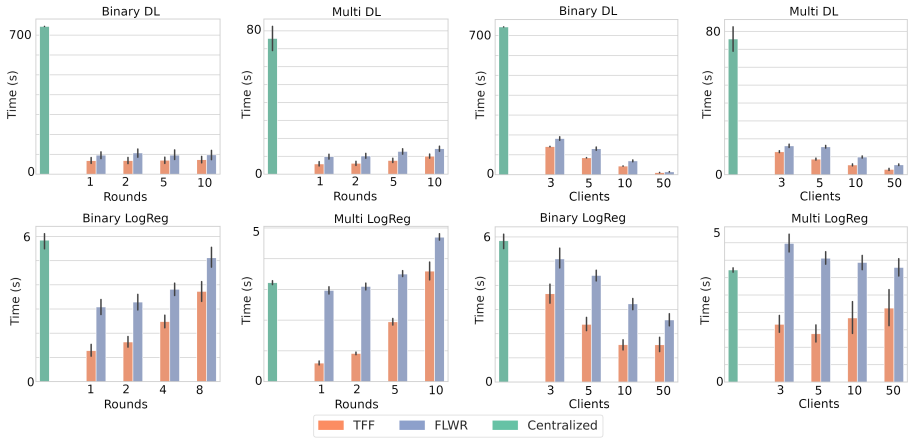

Fig. 5: Local training time with respect to increasing training rounds and clients

Finding 6: FL does not increase individual training times. We measure the training time for each client individually over multiple training rounds. Recall that we keep the total number of training epochs and the total number of samples constant, i.e., the more clients, the fewer local training rounds and the smaller the sample sizes per individual client. Thus, we assume that more clients result in smaller training times per client. Figure 5 confirms this. The figure is organized in the same way as Figure 2, i.e., models in rows and problem types in columns.

FLWR provides a much faster local training compared to TFF, because of differences in their implementation. The difference between centralized training and federated training is less distinct for logistic regression than for training deep learning models.

Thus, healthcare institutions benefit more from federated training for complex machine learning approaches with long training times such as deep learning models. From a global perspective, the increased network traffic (see Figure 7) might slightly increase the total training times. This depends on the number of round and clients, and the network capacity and latency of the coordinator.

Finding 7: Memory consumption is effectively shared. We measured the aggregated memory consumption for both the clients and the overall system.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

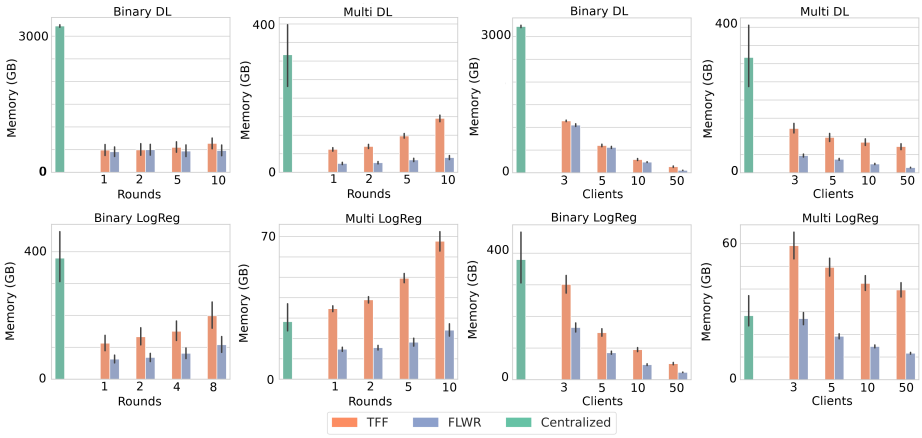

Fig. 6: Local memory usage with respect to increasing training rounds and clients

As the number of clients or rounds increases, the overall resource consumption increases due to the increase in coordination effort. However, the client-wise resource consumption decreases, which is an advantage for healthcare institutions.

We observed that both the clients and the global FL system as a whole require more memory with deep learning than with logistic regression. This was expected, because deep learning uses much more parameters than logistic regression. Second, TFF uses much more memory than FLWR (Figure 6, again with models in rows and problem types in columns). We conclude that FL saves healthcare institutions a significant amount of memory, at the cost of a slight increase in global training time and global memory requirements. Further, FL frameworks show distinct differences in their training times and memory consumption revealing potential for optimization of FL tools.

Finding 8: The network load is not a bottleneck. To assess the network load, we assume that the amount of data transmitted and received is determined by the data serialization method of the framework, and is not influenced by hardware or interference from other clients. Therefore, we experiment with a fixed number of 10 clients. In accordance to the increased memory usage of TFF, TFF comes with higher network traffic as well. Additionally, since deep learning needs to share more parameters than logistic regression, the network traffic rises from 4 MB (peak for LogReg) up to 30 MB (peak for DL).

For comparison, an average household has a bandwidth of 209 Mbps and can easily handle the network demands [11]. We conclude that the network traffic is acceptable for healthcare institutions, and may pose a problem only for the training of very large neural networks such as foundation models. The frameworks have different demands on computing resources, but this is not a basis for selection in most medical scenarios, and it does not restrict the applicability of FL on transcriptomic data.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

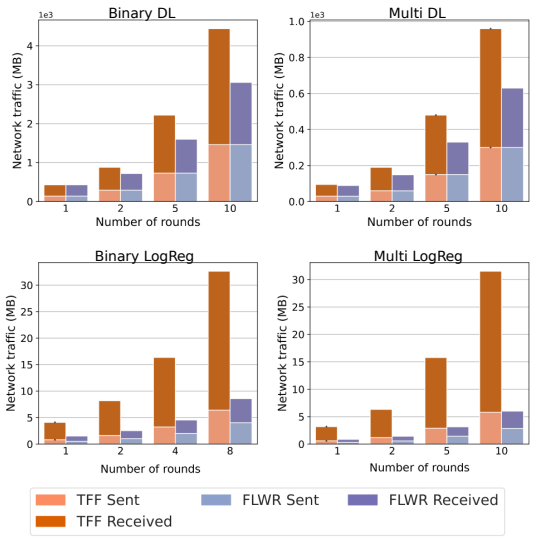

Fig. 7: Network traffic with respect to an increasing number of rounds

5 Conclusions

We have analyzed the challenges of applying Federated Learning to transcriptomic data regarding architectural hegerogeneity, statistical heterogeneity, Gaussian noise and resource consumption. This is important for application areas such as precision medicine, where sensitive patient data is distributed among clients, which do not possess the computational resources for traditional machine learning approaches. In particular, we tested two real-world data sets and use-cases with varying numbers of training rounds and clients, and we compared a centralized baseline with two FL frameworks.

Our analysis shows, that for multi-factorial problems and high number of features, deep learning models outperform logistic regression models in terms of model quality. Increasing the number of training rounds does not greatly improve the global model quality, showing the high effectiveness of weigh aggregation in FL. However, hyperparameter tuning has a large impact. Transcriptomic data is robust to some class imbalance, especially using deep learning models. Problem type and data set are key factors for robustness, also with respect to the amount of training data. Privacy-preserving Gaussian noise can lead to a drastic loss in model quality. FL saves memory and training time for individual clients: The more clients, the lower the individual load. Flower consumes less computational resources than TensorFlow Federated, requires less knowledge about FL, but is also less customizable. The network traffic we measured seems acceptable for typical health institutions. Finally, our findings confirm that FL is applicable and beneficial for disease prognosis and cell type classification using transcriptomic data.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

References

1. tshark: Command line network protocol analyzer. https://www.wireshark.org/docs/man-pages/tshark.html (2024), accessed on: 20.01.2024
2. Abadi, M., et al.: {TensorFlow}: a system for {Large-Scale} machine learning. In: 12th USENIX symposium on operating systems design and implementation (OSDI 16). pp. 265–283 (2016)
3. Adnan, M., Kalra, S., Cresswell, J.C., Taylor, G.W., Tizhoosh, H.R.: Federated learning and differential privacy for medical image analysis. Scientific reports **12**(1), 1953 (2022)
4. Antunes, R.S., André da Costa, C., Küderle, A., Yari, I.A., Eskofier, B.: Federated learning for healthcare: Systematic review and architecture proposal. ACM Transactions on Intelligent Systems and Technology (TIST) **13**(4), 1–23 (2022)
5. Beguier, C., Terrail, J.O.d., Meah, I., Andreux, M., Tramel, E.W.: Differentially private federated learning for cancer prediction. arXiv preprint arXiv:2101.02997 (2021)
6. Beutel, D.J., Topal, T., Mathur, A., Qiu, X., Fernandez-Marques, J., Gao, Y., Sani, L., Li, K.H., Parcollet, T., de Gusmão, P.P.B., et al.: Flower: A friendly federated learning research framework. arXiv preprint arXiv:2007.14390 (2020)
7. Boenisch, F., Dziedzic, A., Schuster, R., Shamsabadi, A.S., Shumailov, I., Papernot, N.: When the curious abandon honesty: Federated learning is not private. In: 2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P). pp. 175–199. IEEE (2023)
8. Choudhury, O., Gkoulalas-Divanis, A., Salonidis, T., Sylla, I., Park, Y., Hsu, G., Das, A.: Differential privacy-enabled federated learning for sensitive health data. arXiv preprint arXiv:1910.02578 (2019)
9. Danek, B.P., Makarious, M.B., Dadu, A., Vitale, D., Nalls, M.A., Sun, J., Faghri, F., Lee, P.S.: Federated learning for multi-omics: a performance evaluation in parkinson's disease. bioRxiv pp. 2023–10 (2023)
10. Dayan, I., Roth, H.R., Zhong, A., Harouni, A., Gentili, A., Abidin, A.Z., Liu, A., Costa, A.B., Wood, B.J., Tsai, C.S., et al.: Federated learning for predicting clinical outcomes in patients with covid-19. Nature medicine **27**(10), 1735–1743 (2021)
11. Fair Internet Report: Internet in the usa - stats and figures. URL: https://fairinternetreport.com/United-States, accessed on: 19.12.2023
12. Fu, L., Zhang, H., Gao, G., Zhang, M., Liu, X.: Client selection in federated learning: Principles, challenges, and opportunities. IEEE Internet of Things Journal (2023)
13. Ganju, K., Wang, Q., Yang, W., Gunter, C.A., Borisov, N.: Property inference attacks on fully connected neural networks using permutation invariant representations. In: Proceedings of the 2018 ACM SIGSAC conference on computer and communications security. pp. 619–633 (2018)
14. Google: Tensorflow federated: Machine learning on decentralized data. https://www.tensorflow.org/federated (25-11-2023)
15. He, C., et al.: Fedml: A research library and benchmark for federated machine learning. arXiv preprint arXiv:2007.13518 (2020)
16. Hodge, R.D., et al.: Conserved cell types with divergent features in human versus mouse cortex. Nature **573**(7772), 61–68 (2019)
17. Hodson, R.: Precision medicine. Nature **537**(7619), S49–S49 (2016)
18. Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al.: Advances and open

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26

problems in federated learning. Foundations and Trends® in Machine Learning **14**(1–2), 1–210 (2021)
19. Kosorok, M.R., Laber, E.B.: Precision medicine. Annual review of statistics and its application **6**, 263–286 (2019)
20. Lee, G.H., Shin, S.Y.: Federated learning on clinical benchmark data: performance assessment. Journal of medical Internet research **22**(10), e20891 (2020)
21. Leo Seeger: Benchmarkingfederated. URL: https://github.com/leoseg/BenchmarkingFederated (2024), accessed on: 20.01.2024
22. Liu, Y., Fan, T., Chen, T., Xu, Q., Yang, Q.: Fate: An industrial grade platform for collaborative learning with data protection. The Journal of Machine Learning Research **22**(1), 10320–10325 (2021)
23. McKinney, W., et al.: Data structures for statistical computing in python. In: Proceedings of the 9th Python in Science Conference. vol. 445, pp. 51–56. Austin, TX (2010)
24. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. pp. 1273–1282. PMLR (2017)
25. Melis, L., Song, C., De Cristofaro, E., Shmatikov, V.: Exploiting unintended feature leakage in collaborative learning. In: 2019 IEEE symposium on security and privacy (SP). pp. 691–706. IEEE (2019)
26. Mendieta, M., Yang, T., Wang, P., Lee, M., Ding, Z., Chen, C.: Local learning matters: Rethinking data heterogeneity in federated learning. In: Proceedings of the Conference on Computer Vision and Pattern Recognition. pp. 8397–8406 (2022)
27. Oestreich, M., Chen, D., Schultze, J.L., Fritz, M., Becker, M.: Privacy considerations for sharing genomics data. EXCLI journal **20**, 1243 (2021)
28. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. the Journal of machine Learning research **12**, 2825–2830 (2011)
29. Pfitzner, B., Steckhan, N., Arnrich, B.: Federated learning in a medical context: a systematic literature review. ACM Transactions on Internet Technology (TOIT) **21**(2), 1–31 (2021)
30. Warnat-Herresthal, S., Schultze, H., Shastry, K.L., Manamohan, S., Mukherjee, S., Garg, V., Sarveswara, R., Händler, K., Pickkers, P., Aziz, N.A., et al.: Swarm learning for decentralized and confidential clinical machine learning. Nature **594**(7862), 265–270 (2021)
31. Warnat-Herresthal, S., et al.: Scalable prediction of acute myeloid leukemia using high-dimensional machine learning and blood transcriptomics. Iscience **23**(1) (2020)
32. Wu, J., Chen, Y., Wang, P., Caselli, R.J., Thompson, P.M., Wang, J., Wang, Y.: Integrating transcriptomics, genomics, and imaging in alzheimer's disease: A federated model. Frontiers in Radiology **1**, 777030 (2022)
33. Zhao, J.C., Sharma, A., Elkordy, A.R., Ezzeldin, Y.H., Avestimehr, S., Bagchi, S.: Secure aggregation in federated learning is not private: Leaking user data at large scale through model modification. arXiv preprint arXiv:2303.12233 (2023)
34. Zhou, J., Chen, S., Wu, Y., Li, H., Zhang, B., Zhou, L., Hu, Y., Xiang, Z., Li, Z., Chen, N., et al.: Ppml-omics: a privacy-preserving federated machine learning method protects patients' privacy in omic data. bioRxiv pp. 2022–03 (2022)
35. Ziller, A., Trask, A., Lopardo, A., Szymkow, B., Wagner, B., Bluemke, E., Nounahon, J.M., Passerat-Palmbach, J., Prakash, K., Rose, N., et al.: Pysyft: A library for easy federated learning. Federated Learning Systems: Towards Next-Generation AI pp. 111–139 (2021)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_26