Privacy – Preserved Covid Forecasting Model

Abstract

While plethora of digital information can drive significant improvements in AI, it is quite challenging to access and process the data in the healthcare industry due to data access and sharing problem, expensive privacy beaches and regulatory reviews. With the help of privacy-enhancing technologies (PETs), these risks can be effectively reduced, and the ideal balance between privacy, security, and compliance can be achieved. In this abstract we present our solution for covid forecasting in a privacy/security preserving manner while optimizing for accuracy using different PETs – FedProx, Differential Private-SGD, Learning with Errors (LWE) Homomorphic Encryption. Our solution provides privacy and security at different layers of data gathering, training, and sharing of model as well as inference from models. Medical data at different institutions/districts may not follow I.I.D and thus we would use featurization and modeling approaches that would perform well for these issues. Our data featurization layer improved our solution's ability to model likelihood of COVID-19 for the patients and ***reduce the chances of an evasion attack***. The next stage focused on training local models using different classification algorithms – Logistic Regression, Random Forest, XGBoost, and NN. Our current solution is currently limited to Logistic Regression since its ease of implementation in Pytorch as compared to Random Forest and XGBoost but these models can be used in a centralized model with ease. A noise vector is additionally added to the data, model, loss function & optimizer by using DP-SGD, which defends against ***privacy inference attacks while maintaining computational resourcing***. Our solution does not propose epidemiological agent-based models (SIR and variants) where we model our infection rate as time varying rate dependent on the mobility and contact network of an individual, since we established through multiple iterations of our classification models and data analysis that COVID-likelihood is a function of infected interaction in the previous 2-3 days of the patient. Model's usage of computational resources is additionally logged to ***detect malicious servers***. Next stage includes sharing gradients/hyperparameters from the various local servers to the central server, where we use LWE to ensure that the weight updates are encrypted with local server key and only the encrypted gradients are shared with the central server ***to counterfeit model poisoning attacks***. Though this increases the size of the encrypted parameters and computational cost of the sharing process, we ***use model compression approaches to increase calculation efficiency***. LWE has scalability issues where we are limited by the current hardware specifications of the challenges but there is a scope of faster and compressed encryption of gradients. We implemented both FedProx and FedAvg and hence propose using ***FedProx instead of FedAvg*** since medical data may not have I.I.D as assessed during the data validation stage, and FedProx tends to perform better than FedAvg under such scenarios, hence making our solution provide ***robust aggregation***. We will incorporate data quality assessment score while weighing the gradients/hyperparameters for local servers. The updated central-server global model is shared with local servers (in case of FL) to improve the global model to provide defense against different attacks ***like model inversion & model extraction*** which we aim to reduce by ***controlling overfitting***. We believe that our employed secure standards for data and model communication our solution will provide value to different healthcare organizations and government institutions. Current restriction of Flower Framework limited our experimentation with Swarm learning and other privacy enhancing technologies.

Background

The convergence of smartphones, cloud, and AI-based technologies has generated unprecedented opportunities for the development of data-driven insights that can be used for betterment of patient health. However, unlike other industries where AI has been applied, access to sensitive healthcare data continues to be a challenge in healthcare owing to multiple privacy and security reasons. Development of AI solutions using such sensitive healthcare data will necessitate novel approaches that address such concerns, while at the same time being accurate enough to be of practical utility.

Privacy Enhancing Technologies (PETs) offer a way out here – allowing development of machine learning models/ability to glean insights from sensitive data; without accessing it directly and keeping it under the control of its owners. A range of PETs such as Homomorphic Encryption, Differential Privacy, Secure Multi-Party Computation (SMPC), Zero Knowledge Proofs are available and can be readily used; many other PETs continue to be an active area of research.

The recent COVID-19 pandemic has brought to fore the need for such solutions to address a range of problems; primary among which is producing appropriate disease prevention/intervention plans based on local situations. Such plans necessitate developing models that can predict the likelihood of a patient to develop an infection, which can then be utilized to develop bespoke curative and prevention tactics at an appropriate geographical level. Our solution aims at developing such privacy-preserving, federated machine learning models, utilizing differential privacy (DP)/Homomorphic Encryption (HE) methods that can be used to predict the individual model of infection.

Threat Model

Attack	Mitigation (in proposed Solution)
Model Inversion	Differential Privacy, Featurization (Model Design)
Membership inference	Differential Privacy
GAN Reconstruction	Differential Privacy
Model Poisoning	Learning with Errors, Model monitoring, Differential Privacy
Aggregation service	Differential Privacy, Learning with Errors

Table 1. Different attacks mitigated by our solution

While developing the federated learning framework we have categorized the threats (Table 1.) associated with FL into two broader categories and their corresponding solutions:

- Ensure Data Privacy by Private AI (Differential Privacy)
- Ensure Model performance with Secure AI (Federated Learning, Homomorphic Encryption)

Ensure Data Privacy by Private AI

Federated learning enables cooperative machine learning without transferring data sets to an external service, it does not prevent information leakage from the model parameters. Machine learning models are generally vulnerable to attacks intended to extract information about the training data via interaction and the analysis of a trained model’s parameters. The major inference-based attack vectors are – 1) Model Inversion attack (extracting sensitive properties of the classes and/or individual samples presented by the model). 2) Membership inference (determining samples that are present in the training data). 3) GAN reconstruction (reconstructing the training data by a synthetic data with similar distributions as training data)
To protect our system from these inference attacks, we have used Differential Privacy, which by adding sufficient noise (Local DP, considering) can reduce the chances of membership inference attack, since it would make it difficult to distinguish between samples that were present vs. the ones that were maliciously added to track membership. Further, adding noise reduces the likelihood of extracting information regarding the model classes or samples. Since we are using featurization that creates additional features than the raw data, it acts as an empirical defense against the model inversion and member inference attacks.

Ensure Model performance with Secure AI

Model performance attacks occur at training phase as well as inference stage of the model, which affect the overall performance of model that leads to a corrupted or a degraded model. During training phase, the attacker can run data poisoning attacks to compromise the integrity of training dataset collection by Label Flipping, Perturbations, etc., or model poisoning attacks (Gradient manipulation, training rule manipulation) to compromise the integrity of the learning process. These attacks can hamper both the privacy of the model as well as the utility.

We implemented Differential Privacy and Learning with Errors which minimize the risk of information leakage from model parameters/outputs thus preventing backdoor insertion attacks. Differential privacy also reduces the extent of overfitting in the model, which leads to a well generalized and robust model. A generalized model tends to be more privacy-preserved as the rate of “memorization” is less which is prone to information leakage via attacks. Thus, not only it may provide a robust model performance but also added good privacy to our solution.

We assume that the central aggregator is honest-but curious server, which may try to gain additional information which is further limited by Differential Privacy and Learning with errors, which by adding noise and encryption on the model parameters, removes chances of parameter leakage to the aggregator.

Technical Approach

Our solution (Figure. 1) is designed as a horizontal Federated Learning approach where there is a central server which will be interacting with other local servers (federation units) to iteratively develop a model that can forecast the likelihood of COVID-19 for a patient. The central server is the orchestrator that will select the number of federation units that will participate in a single round of training. Once the federation units are selected based on the criteria and their corresponding fraction of data to be used for modeling is decided by the central server, they are instantiated for training and communication with the central server. In the first round, all the federation units are initialized with Neural networks, (but may) change based on most high performing models for the federation units. These individual Neural Networks are used for developing local models which are further shared with central server to update the global model.

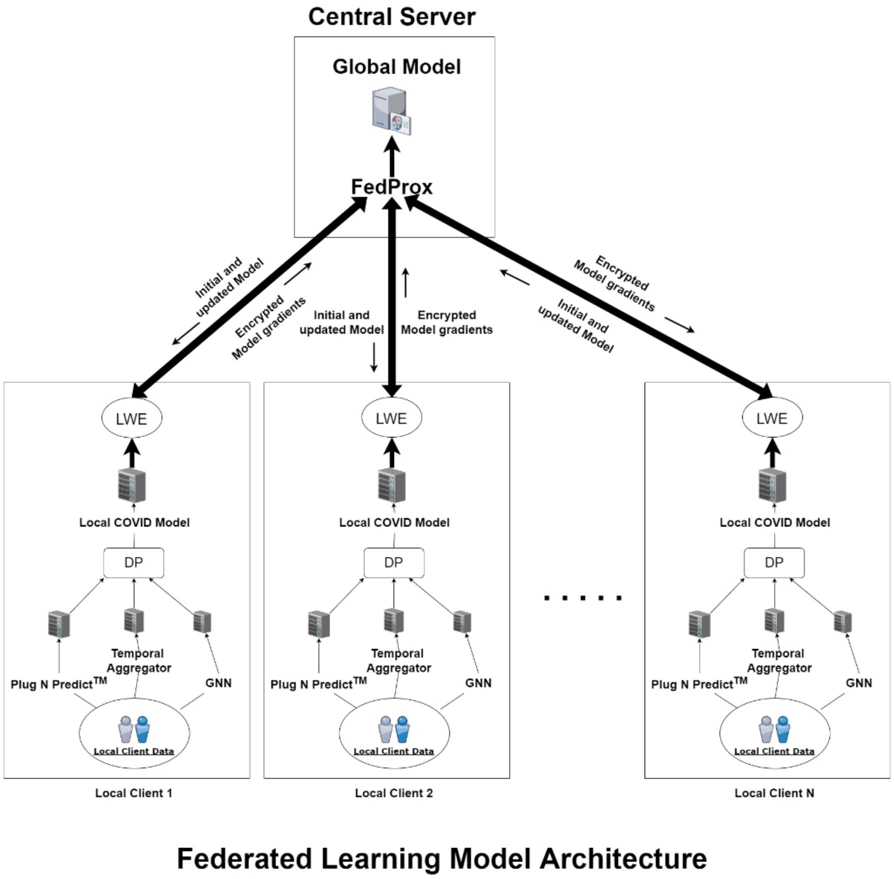

Central Server
Global Model
FedProx
Initial and updated Model
Encrypted Model gradients
Initial and updated Model
Encrypted Model gradients
Encrypted Model gradients
Initial and updated Model
LWE
Local COVID Model
DP
Plug N PredictTM
Temporal Aggregator
GNN
Local Client Data
Local Client 1
Local Client 2
Local Client N
Federated Learning Model Architecture

Figure 1. Solution Architecture

Once the federation unit is selected, following steps are done:

Data Processing

In the data authoring phase, we will run a series of quality checks on data in each of the initialized notes to assess the degree of missingness, uniqueness, completeness, heterogeneity (non-IID), as well as add noise to introduce privacy.

Handling non IID data

Non-IID can typically result from an imbalance in the amount of data, features or labels; this is quite a common occurrence in the medical domain given the differences in data acquisition, instrument calibration, resource limitations etc. As such, any federated learning solution needs to actively assess and come up with mitigation measures for the same.

Featurization

Data processing:

Medical data suffers from low capture rates and data availability issues, hence there is a need to pre-process the data before using it for training as well as inference. The data model requires us to use different constraints (e.g., unavailability of contact information due to different geographies of federation units) among the individual datasets before merging them to create a comprehensive database(s), as the entire data will not be available due to Horizontal Federated learning. Each federation unit is required to follow the constraints related to identifiers of individual persons to limit the access control to Household, Residence and Contact network information. The data is further cleaned for removing any inconsistency and outliers, which may affect the performance of the local model.

We propose feature generation to remove direct dependence of model and raw data, which can reduce the extent of data retrieval if the model is compromised.

Feature generation:

- Plug N Predict - Plug-and-predict is a platform for automated feature engineering, discovery, and machine learning modeling at scale by ZS Associates. Leveraging evolutionary algorithms and ensemble models, plug-and-predict sifts through the high-dimensional search space of features and models to figure out the best possible features and model for your prediction problem automatically. The only input needed are transactional data and the specifications for the covid-forecasting problem. We couldn’t use our propriety platform due to certain limitations associated with the competition.
- Aggregator/Temporal based features – We create temporal features based on the count and moving averages for number of interactions & interaction types, duration of previous interaction and average interaction duration in the previous week/month. These features can encapsulate the information required by a supervised learning algorithm to model covid likelihood based on our features. We also aggregate the disease count as well as other health related outcomes to embed the medical history in hand-crafted features. Since these features are hand-crafted and rule based, they might suffer from getting exposed if someone compromises with other local servers. Hence, to use embeddings for networks that can additionally when used improve the learning process.

The feature generation approaches ensure that the created features are interpretable, such that even after using Privacy Enhancing technologies, that would mask the data as well as the model, the explanations can be generated at the federation unit level. We do not propose using explanation methods like LIME and SHAP, that provide individual explanations and instead use global-level explanations. These explanations can still be used for analyzing infection rate at the federation unit (or sub-unit) level, but at individual level would exploit our solution to attacks, hence we don’t use it. These were further considered to be out of scope for the current solution.

Our assumptions and corresponding featurization is further explained in Figure 2., where we have determined the look-back & look-forward periods to properly identify covid and non-covid patients from the data. This strategy of considering a look-back and look-forward period helped our model to improve accuracy over both centralized and federated solutions. Based on this we have come up with list of features in Table 2.

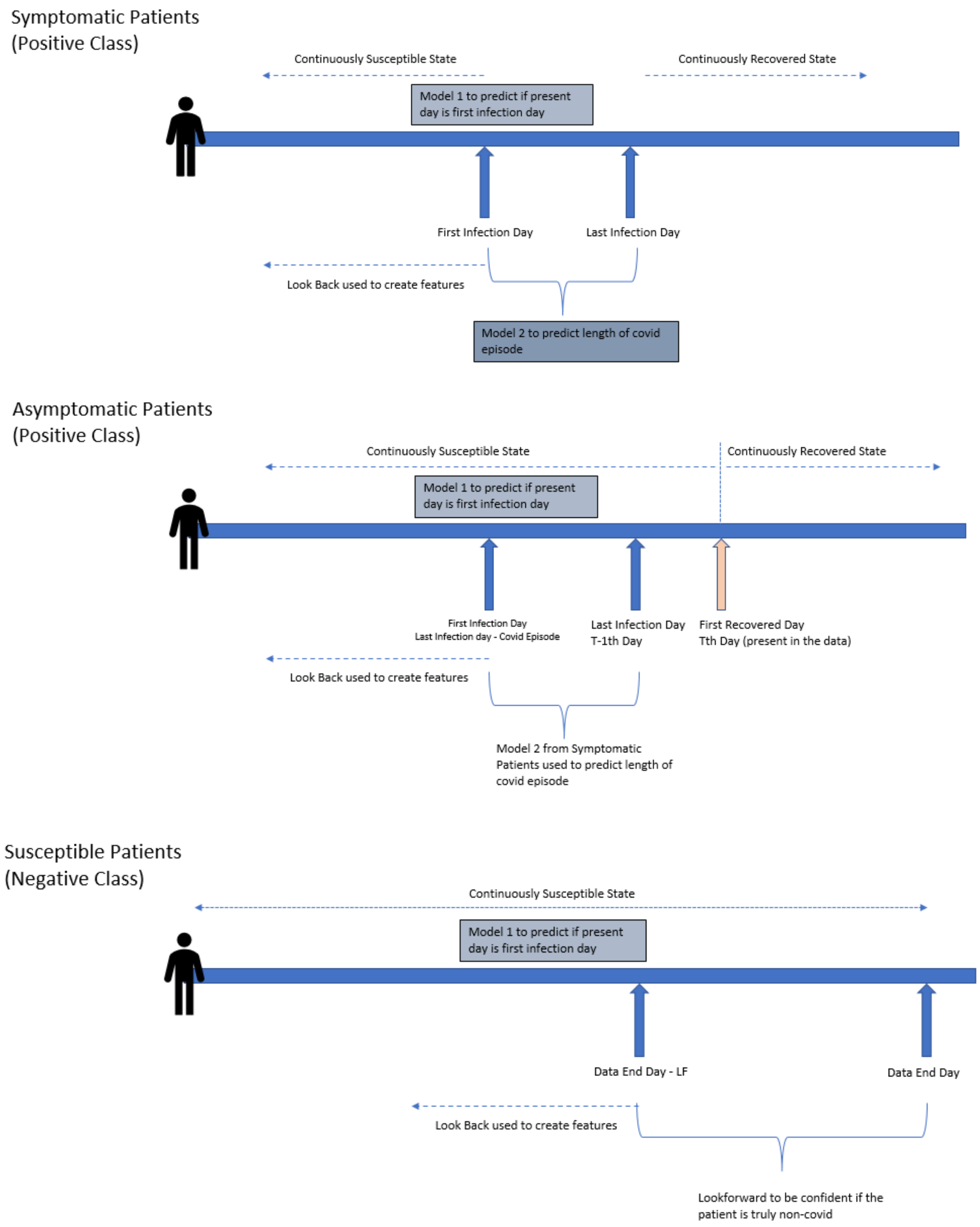
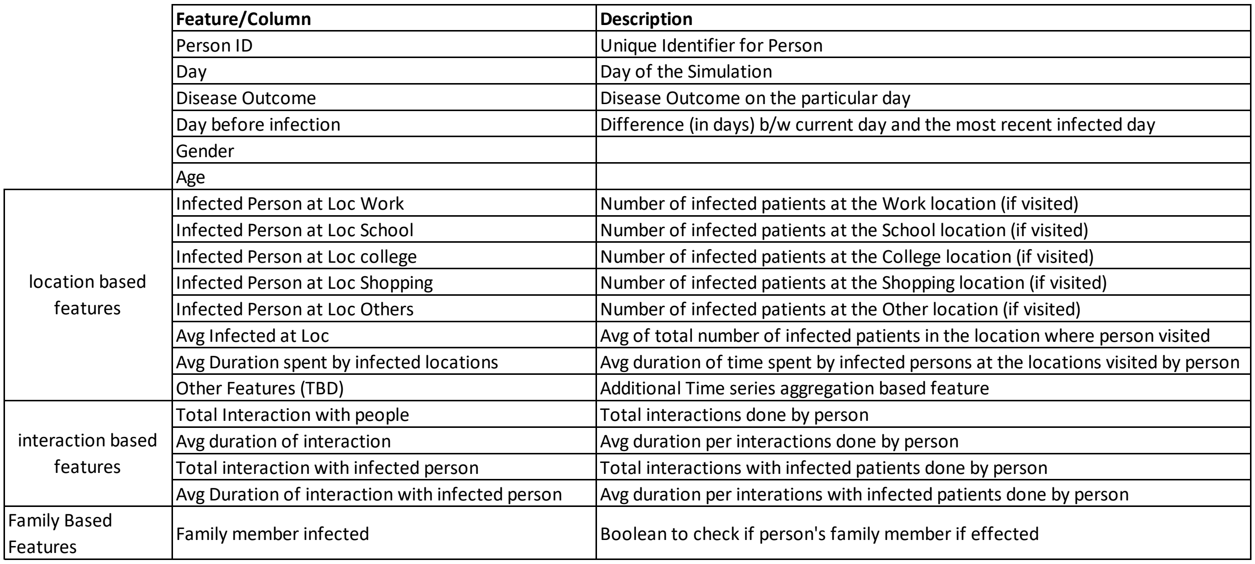

Symptomatic Patients (Positive Class)
Continuously Susceptible State
Continuously Recovered State
Model 1 to predict if present day is first infection day
First Infection Day
Last Infection Day
Look Back used to create features
Model 2 to predict length of covid episode
Asymptomatic Patients (Positive Class)
Continuously Susceptible State
Continuously Recovered State
Model 1 to predict if present day is first infection day
First Infection Day
Last Infection day - Covid Episode
Last Infection Day
T-1th Day
First Recovered Day
Tth Day (present in the data)
Look Back used to create features
Model 2 from Symptomatic Patients used to predict length of covid episode
Susceptible Patients (Negative Class)
Continuously Susceptible State
Model 1 to predict if present day is first infection day
Data End Day - LF
Data End Day
Look Back used to create features
Lookforward to be confident if the patient is truly non-covid

Figure 2. Feature Generation



	Feature/Column	Description
	Person ID	Unique Identifier for Person
	Day	Day of the Simulation
	Disease Outcome	Disease Outcome on the particular day
	Day before infection	Difference (in days) b/w current day and the most recent infected day
	Gender	
	Age	
location based features	Infected Person at Loc Work	Number of infected patients at the Work location (if visited)
	Infected Person at Loc School	Number of infected patients at the School location (if visited)
	Infected Person at Loc college	Number of infected patients at the College location (if visited)
	Infected Person at Loc Shopping	Number of infected patients at the Shopping location (if visited)
	Infected Person at Loc Others	Number of infected patients at the Other location (if visited)
	Avg Infected at Loc	Avg of total number of infected patients in the location where person visited
	Avg Duration spent by infected locations	Avg duration of time spent by infected persons at the locations visited by person
	Other Features (TBD)	Additional Time series aggregation based feature
interaction based features	Total Interaction with people	Total interactions done by person
	Avg duration of interaction	Avg duration per interactions done by person
	Total interaction with infected person	Total interactions with infected patients done by person
	Avg Duration of interaction with infected person	Avg duration per interations with infected patients done by person
Family Based Features	Family member infected	Boolean to check if person's family member if effected

Table 2. Updated feature list for centralized* and federated model

**few difference based features are considered in addition for centralized model*

Addition of Noise (Differential Privacy)

We further add Laplacian Noise to the processed dataset with additional features from Contact network, Demographic and Disease history. This reduces our computation budget while keeping privacy in regards of the solution. Irrespective of this, we will also experiment around different mechanisms will also include a comparison for scalability, as certain mechanism would perform for large datasets (1 billion records) whereas some may perform well only for smaller datasets.

Local Model Training

The models are further trained on the local server adhering to the computational requirements of the server. We propose using Differential Private SGD that would add noise to the loss function and optimizer, and since the training data does contain noise from Exponential mechanism, the entire training loop would be Differentially Private. The training epochs are subject to the computation environment, and we would test the use of Parallel version of Differential Private SGD to improve the training efficiency. Both Gaussian and Exponential mechanism would be evaluated in terms of privacy budget and computational requirement before finalizing the local models. We are hereby assuming that our choice of baseline model would be a Neural network (*adaptation of Logistic Regression*), where can try approximating the function for Covid forecasting.

We will be logging the computational usage as well as the local model performance metrics over time and analyze the information for detecting any malicious use. Sudden increase/decrease in model performance or computational usage can alarm in case of an attack to the server. This would prevent our model from getting compromised in the distributed FL computation.

Using our current mechanism of Differential Privacy also enables us to scale it across high cardinality data, whereas some can expand our solution space to high dimensionality. We believe our solution would be scalable as well as feasible with sufficient compute power, to larger number of districts as well as countries.

Global Model Training

Once the selected local servers have trained their models, the model parameters are aggregated with FedProx which performs better than FedAvg for non-i.i.d data which is a common nuance with medical data. FedProx can help us aggregate these gradients in a better way, since FedProx uses regularized loss with a proximal term. This leads to providing robust aggregation to our models, we will further experiment with other adaptive aggregation mechanisms, if they can provide both better privacy and utility. FedProx also helps the slower clients to contribute to the training with a reduced number of epochs reducing the overall computation complexity and workload at the central server, thus increasing our solution efficiency, and improving scalability. Since we are using a regularization term here, this will also reduce the overfitting problem. Robust models are less prone to attacks since they generalize better and prevent the models from targeted attacks.

Once the model weights are aggregated along each round and assessed for performance and fairness metrics, they are shared across all the local servers for inference for new patients. The model performance would be comparable to the central non-private baseline model as our privacy measures employ using Differential Private SGD, which though reduces the analytical utility by increasing the existing biasness in the model (Bagdasaryan, 2019).

Proof of Privacy

Our solution uses Differential Privacy and Learning with Errors approach to provide additional privacy and security to the Horizontal Federated Learning framework. The differential privacy parameter would have a privacy budget of ε (epsilon), which quantitatively governs the privacy of the differential private mechanism. Since differential privacy is focused towards adding noise, it would reduce the utility of the entire solution to some extent. The privacy and utility have a trade-off which is primarily dependent on ε and the scale-parameter (1/ε), with different features and algorithms there is a need to update values for scale-parameter to ensure ε private approach. This scale parameter's values are updated based on underlying mechanism for differential privacy – Laplacian, Exponential and Gaussian. Advanced approaches like matrix mechanism can significantly improve the utility of the solution while keeping the privacy budget same. The privacy vs. utility trade-off can further be expanded by comparing the raw and differential private data on summary statistics (like mean absolute error, root mean squared error, coefficient of variation, etc.), outcome specific analysis, marginal distributional metrics (Chi-square test, Kolmogorov-Smirnov test) and global utility metrics (Bowen, 2021). These metrics can help control the epsilon and scale parameter at each federation unit to ensure there is a balance of both privacy and utility. There are newer variants of DP like Rényi DP and zero-concatenated DP, which may provide better privacy to our solution by appropriate averaging of privacy budget across different databases/outputs. We believe using $\varepsilon > 1$, would also be useful in providing higher privacy guarantees (as used by some of the published parameters (Desfontaines, 2021))

Experimental Results

Privacy: Privacy is defined by the parameters used by our privacy techniques. For the solution submitted, we have used FedProx where mu = 0.01 and Differential Privacy where our parameters are ε (privacy budget) =3 and $\delta=1/n_c$ where n_c is the number of samples for the client

Accuracy: Area under the precision-recall curve (AUPRC) was noted for every iteration

Efficiency: For efficiency we have noted the total execution time, computation time and overall memory usage

Scalability: Measured by the change in execution time and AUPRC score as number of partitions/clients increases

The privacy-accuracy tradeoff was also observed and measured during execution of multiple variations in different parameters.

Here are the experimentation results -

Scalability	Privacy	Accuracy		Efficiency			
Number of Clients	**Epsilon**	**Loss**	**AUPRC**	**Execution Time (sec)**	**Computation Time(sec)**	**% RAM used**	**Memory Usage**
3	5	0.91	0.032	292.32	1.28	50.7%	33.12GB
3	3	0.900	0.032	279.14	1.32	50.7%	33.14GB
3	1	0.8884	0.0402	279.35	1.259	50.7%	33.115GB
3	0.003	1.022	0.03723	278.84	1.304	50.0%	32.70GB
5	3	0.74409	0.0321	727.14	2.875	50.7%	33.127
10	3	0.79	0.033	1357.68	5.45	50.4%	32.96GB
Centralized	-	0.78	0.041	237	0.89	50.43%	33.16GB

We observed that both moderate and strict criteria of privacy budget showed similar performance I.e., AUPRC scores were very close. With increasing clients and exposure to more data the AUPRC improved with strict privacy budget. The RAM usage and memory used remained similar despite the time taken increased with increasing number of clients.

Data

For the COVID forecasting track of the competition, only one modality of data i.e., tabular data is provided in three different forms – Person-level static data (demographics and residence data), Network graph (social contact network data) and temporal data (activity and disease outcome data). To handle these diverse data sources, we have proposed different data featurization architectures to create person-level features which are feed into the local forecasting models.

Discussion and Next Steps

We could not implement certain technologies that we proposed in our solution during the competition Phase 2 timelines based on infrastructure, implementation and time constraints but can be added to increase privacy and security while maintaining the privacy -

<u>Query Control Mechanism:</u> The attacker can also launch a range of inference attacks on an individual participant’s update or on the aggregate of updates from all participants. Attacks at inference phase are called evasion/exploratory attacks. They generally do not tamper with the targeted model, but instead, either cause it to produce wrong outputs (targeted/untargeted) or collect evidence about the model characteristics. The effectiveness of such attacks is largely determined by the information that is available to the adversary about the model. Inference phase attacks can be classified into white-box attacks (i.e., with full access to the FL model) and black-box attacks (i.e., only able to query the FL model). Our query control mechanism tracks the # queries made by each federation unit, and there is a quota for the maximum # of queries. Since, based on our specific use case, we do not expect many queries from the federation units, we propose using a smaller number of queries. This further reduce the communication bandwidth and make our solution more feasible and scalable. We did not have required exposure to the infrastructure where we can implement query control mechanism and thus we had to remove it from our implementation.

Ongoing data trends: The data at each federation unit is analyzed to understand the data as well as concept drift. These are commonly assessed metrics for operationalized model development, but in our case, they will be more useful to spot sudden/inconsistent drifts. Sudden drifts need to be debugged to understand if there is an attacker or the distribution of the underlying data and the relationships between the patient's metric and COVID-19 likelihood have changed. The distribution metrics like summary statistics are analyzed over time to see if there are any changes during the time. An outlier-detection algorithm will help us detect any poisoning attacks for the data, in case of attacks the local server would be prohibited from accessing computational resources, local models and local data.

The central server will also review these data quality logs to distinguish if a certain local server is performing differently as compared to other local servers, leading to distinguish a corrupt or malicious server. The data assessment pipeline is the empirical defense that ensures data privacy by design in our solution.

Homomorphic Encryption (Learning with Errors): Since, no existing integration of LWE exists with Flower framework, our own script had run-time constraints due to parallelization. Hence, we didn't use it but it can help significantly improve the model security and privacy.

Team Introduction (ZS_RDE_AI)

Based in Washington DC, Dr. Qin Ye is the Global RWE lead for ZS. Qin joined ZS in 2016, where he has been focusing on helping clients realize the value of their RWD investment in R&D through our strategy, data science, and technology capabilities. In his role, Qin leads the development of our RWE team, oversees the development of our innovative RWE offerings, and works directly with clients on evidence generation and advanced data sciences programs across clinical, HEOR, medical, and market access functions. As a trained physician, outcome researcher and data scientist, Qin has over 20 years of health informatics and data analytics experiences with in-depth knowledge of medical terminologies, data standards, various EHR products, and wide range of RWD use cases throughout life sciences product lifecycle, e.g., pragmatics trials designs, observational study methods and actionable insights. Qin has a distinctive combination of patient care, health informatics, life science data analytics and leadership experiences across the healthcare industry. These include Director, Health Informatics at AstraZeneca where he led the development of global and cross-functional RWE technology and analytics capabilities; Chief Medical Informatics Officer at Acupera; Executive VP at Medversant Technologies; Associate Medical Director at Pfizer; and Principal Architect at IDX Systems Corp.

Mayank Shah is a Data Science Consultant at ZS and holds a Bachelor's in Technology for Computer Science from Maharaja Agrasen Institute of Technology. He is an experienced data scientist with experience in Medical Imaging AI, Biomedical NLP, and Statistical analysis with interests in improving clinical trials with AI.

Shaishav Jain is a Data Science Associate Consultant at ZS. He has a Bachelor's in Electrical and Electronics Engineering from the Vellore Institute of Technology. He has experience working on different data modalities and problem fields like Medical Imaging AI, NLP, and Patient Identification using EHR data.

Sagar Madgi is an Associate Principal with the R&D Excellence team in ZS. In this current role, he's specifically focused on building AI products as well as executing be-spoke AI projects leveraging both clinical and real-world patient data sources. He is experienced in the use of multiple real world data source (claims, EHR) and advanced data science to build products as well as execute be-spoke projects to produce actionable insights across R&D and commercial in the life sciences industry.

References

A. N. Bhagoji, S. C. (2019). Analyzing federated learning through an adversarial lens. *Proc. Int. Conf. Machine Learn.*, (pp. 634-643).

Ali-Ozkan, O., & Ouda, A. (2016). A Classification Module in Data Masking Framework for Business Intelligence Platform in Healthcare. *IMECON*.

Bagdasaryan, E. a. (2019). Differential Privacy Has Disparate Impact on Model Accuracy. *Proceedings of the 33rd International Conference on Neural Information Processing Systems.* Red Hook, NY, USA: Curran Associates Inc.

Bowen, C. M. (2021, November 29). *Utility Metrics for Differential Privacy: No One-Size-Fits-All*. Retrieved from NIST.gov: https://www.nist.gov/blogs/cybersecurity-insights/utility-metrics-differential-privacy-no-one-size-fits-all

Desfontaines, D. (2021, 10 01). *A list of real-world uses of differential privacy*. Retrieved from Ted is writing things (personal blog): https://desfontain.es/privacy/real-world-differential-privacy.html

Jie Zhou, G. C. (2020). Graph neural networks: A review of methods and application. *AI Open*, 57-81.

Peikert, C. S. (2019). Noninteractive Zero Knowledge for NP from (Plain) Learning with Errors. *Advances in Cryptology* -- *CRYPTO 2019* (pp. 89--114). Cham: Springer International Publishing.