




OPEN

## CoVnita, an end-to-end privacy-preserving framework for SARS-CoV-2 classification

Jun Jie Sim , Weizhuang Zhou, Fook Mun Chan, Meenatchi Sundaram Muthu Selva Annamalai, Xiaoxia Deng, Benjamin Hong Meng Tan & Khin Mi Mi Aung

Classification of viral strains is essential in monitoring and managing the COVID-19 pandemic, but patient privacy and data security concerns often limit the extent of the open sharing of full viral genome sequencing data. We propose a framework called CoVnita, that supports private training of a classification model and secure inference with the same model. Using genomic sequences from eight common SARS-CoV-2 strains, we simulated scenarios where the data was distributed across multiple data providers. Our framework produces a private federated model, over 8 parties, with a classification AUROC of 0.99, given a privacy budget of  $\epsilon = 1$ . The roundtrip time, from encryption to decryption, took a total of 0.298 s, with an amortized time of 74.5 ms per sample.

The pathogenic virus known as SARS-CoV-2 emerged quietly in the final months of the year 2019 in Wuhan, in the Hubei province of China. The virus caused severe respiratory symptoms in patients and was highly transmissible, spreading across the globe within weeks. By 31st January 2020, the World Health Organization had declared a public health emergency on COVID-19, the disease caused by SARS-CoV-2. As the third pandemic of the 21st century, COVID-19 placed significant stress not just on the global healthcare infrastructure, but also on the global supply chain networks. Although significant discoveries in COVID-19 vaccines and anti-viral treatment options have helped restore a semblance of normality in several developed countries, COVID-19 remains a threat globally two years into the pandemic. Part of the difficulty in stemming the tide, or eradicating the virus, is due to the high mutation rate of the virus. The pandemic has been marked by waves of new infections and reinfections caused by the rise of newer strains of the virus: first the alpha and beta strains at the end of 2020, which were then displaced by the more transmissible delta strain, and eventually the current dominant strain, omicron. Given the speed at which novel coronavirus variants develop in different geographical pockets around the world, there is a pressing need for the development of infrastructure to perform global surveillance and outbreak prediction. To this end, initiatives such as the Global Initiative on Sharing Avian Influenza Data (GISAID) have played an important role in monitoring the pandemic.

Yet, privacy concerns have been raised regarding the sharing of viral sequencing data, as such data can be used in tandem with other contextual clues to establish patient identity. As the virus spreads mainly by close contact, the circulation of a new viral strain within groups of individuals may indicate that they have engaged in social activities together. This can lead to ostracization or discrimination, especially if there is a social stigma associated with the disease within the community. For instance, one of the COVID-19 clusters in South Korea had been linked to members of a religious cult<sup>1</sup>, while one of the largest COVID-19 waves in Singapore was linked to socializing between karaoke patrons and sex workers<sup>2</sup>. In both cases, the patients were accused of not following public health guidance in place at that time, and for acting irresponsibly and selfishly. Although they were not the only patients who had contracted COVID-19 during that period, they could be identified because of the strains they were carrying. To assuage the privacy and security concerns of institutions and individuals regarding such crucial information, care must be taken to ensure that the confidentiality of the data is not compromised during any downstream analysis which is achieved with our proposed framework, CoVnita, by obfuscating samples and the classification outcome with HE, preventing unintended negative consequences.

The Integrating Data for Analysis, Anonymization and SHaring (iDASH) centre organizes the annual Secure Genome Analysis Competition to bring together bioinformaticians and cybersecurity experts to address problem statements regarding privacy concerns arising from data sharing in bioinformatics research. In 2021, Track II of the competition posed the pertinent challenge statement: to detect and track viral strains, SARS-CoV-2 viral

Institute for Infocomm Research, Agency for Science, Technology And Research (A\*STAR), Singapore, Singapore.  
✉ email: simjj@i2r.a-star.edu.sg

samples have to be classified as one of many known strains. However, the sample data cannot be shared due to policies put in place to protect patient privacy.

In this paper, we present a framework to first enable the private training of a model and then secure classification using the said model. Note that the secure classification technique described here was the winning solution submitted by our team (A\*FHE-2) to Track II of the iDASH 2021 competition<sup>3,4</sup>. Beyond demonstrating the analysis of genomic data privately and securely, we have also performed additional analysis to assess the eroding effect of differential privacy and data variability on model performance. Our proposed framework achieves the following:

- Enables different organizations to jointly train a model securely end to end, preserving the privacy and confidentiality of patients' data from training to inference.
- Provides quick and accurate classification of COVID-19 strains to improve triage of patients based on the predicted outcomes of the classified strains, thereby alleviating the burden on hospital infrastructure.

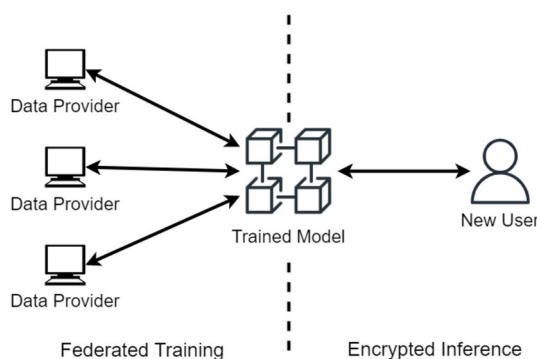
## Results

CoVnita, Fig. 1, provides an end-to-end workflow that trains a model across multiple parties securely with Federated Learning (FL), further reinforced by injecting Differential Privacy (DP), and classifies new samples privately with Homomorphic Encryption (HE). An honest-but-curious threat model is assumed in this work. This means that the parties in the protocol will adhere to the protocol, but are curious about another party's private information. All communication channels between parties are assumed to be secure.

**Privacy-preserving model training.** We demonstrate our framework on eight COVID-19 variants that had gained prominence at various time points during the pandemic, including four that were provided in the iDASH 2021 competition<sup>4</sup>. We characterize the quality of the genomic data from these sequences in Table 1.

The sequences were reduced to a more compact set of hashed features using a technique called Dashing (see "Methods"). For more efficient training and inference, we perform a round of federated feature selection using Fed- $\chi^{25}$  to approximate the top 15 informative features. Then, we apply FL to enable data contributors to jointly build the virus strain classification model without revealing their data. DP was applied to further enhance the privacy of the dataset by releasing differentially-private local models

Two setups, a balanced and imbalanced data split, were used to evaluate the model training framework. The first setup involved balanced combinations where the data is split evenly over 8 parties in various degrees, ranging from 1 up to 8 variants per party. There are multiple permutations for each scenario where each party hold either 1, 2, 4 or 8 variants, with the same number of samples per party (2000). One possible combination is described in Supplementary Tables S1 to S4. This results in differing local models that aggregate into distinct global models. A total of 301 possible configurations were tested, 100 for each of the scenarios with 1-, 2- or 4-variants and 1 for the 8-variant case. These were generated randomly and the average performance for the different variant scenarios was recorded. The second setup considered two different types of imbalanced data splits that may be more applicable to real-world scenarios. The first imbalanced data split focuses on the 4 variants, B.1.617.2 (Delta), C.37 (Lambda), B.1.621 (Mu), B.1.1.529 (Omicron), obtained from the GISAID database, based on their preponderance in different geographical continents, which we described in Supplementary Table S5. In



**Figure 1.** Outline of CoVnita. The training phase begins with the data providers each locally training a differentially private local model. A federated feature selection is first performed to reduce the size of the data while maintaining its quality. Next, they share and compute a global average of their local models. This process of local updates and joint averaging then repeats for a fixed number of epochs. Throughout this process, a joint global model, that does not require data providers to share their genomic samples, is trained. To ensure the robustness of the model against membership inference attacks, differentially private noise is injected during the training process. The classification process is performed in its entirety with the samples encrypted with HE, a special type of encryption that supports computation on encrypted data. This means that new samples can be evaluated in their encrypted form, ensuring that the data stays private. An efficient HE packing method was used, allowing many samples to be simultaneously classified.

PANGO lineage	Variant	A	T	C	G	N	Others
B.1.617.2	Delta	17,350,073	18,673,551	10,640,782	11,396,199	1,435,543	1618
C.37	Lambda	17,678,438	19,008,782	10,830,235	11,585,368	404,444	1522
B.1.621	Mu	17,324,855	18,658,329	10,624,188	11,385,289	1,439,488	1168
B.1.1.529	Omicron	17,359,295	18,659,613	10,643,562	11,405,345	1,426,693	551
B.1.429	Epsilon	17,658,000	18,978,000	10,838,000	11,596,000	494,000	0
B.1.1.7	Alpha	17,698,533	19,045,276	10,868,253	11,617,247	404,750	542
P.1	Gamma	17,740,786	19,086,620	10,892,220	11,648,508	300,750	417
B.1.526	Iota	17,699,086	19,043,095	10,867,848	11,618,684	421,301	326

**Table 1.** Distribution of nucleotide bases. The percentage of uncertain bases (*N* and *Others*) in the sequences ranges from 0.5 to 2.5%, with the Gamma and Mu variants having the highest and lowest quality sequences respectively.

the second imbalance data split, described in Supplementary Table S6, the data was randomly assigned amongst the 8 parties via sampling from a uniform distribution.

To demonstrate the effectiveness of our framework, we also trained a model with the entire dataset to use as the baseline for comparisons with models trained without DP. We present the performance of our models trained with a total of 16,000 samples, 2000 samples per variant. The models were tested with an unseen test set of 4000 samples—500 samples per variant. These results are summarized in Table 2 and their statistical distribution is given in Supplementary Tables S7 to S9. Performance (based on AUROC) between models trained in centralized and federated settings was largely similar.

The models were then subsequently enhanced with DP and tested with the same samples. Increasing the amount of noise introduced during the DP-SGD training process leads to a lower privacy budget. Table 3 describes the federated model performance over four different privacy budgets  $\epsilon = 0.1, 1.0, 3, \infty$ , in decreasing order of privacy.

The geographical split scenario produced models with the lowest model accuracy (0.338, Table 3) due to the asymmetrical split of the data, and a stronger privacy guarantee in this scenario resulted in larger distortion on an already sparse data split. However, the AUC metric remains relatively high (0.710, Table 3), indicating that the classification of certain variants remains fairly accurate.

Overall, our results suggest that FL with DP is a feasible approach to enable privacy-preserving collaborative machine learning in real-world settings.

**Homomorphic classification.** We used the SEAL library (version 3.2.2)<sup>6</sup> for HE instantiation with the following parameters:  $\log N = 13$ ,  $\log p = 30$ ,  $\log Q = 90$ . This gives us a security level of at least 128-bits according to the estimator by<sup>7</sup>. This particular setting allows a ciphertext to support up to 4096 samples. The same test set of 4000 samples that were used to evaluate the federated models was encrypted and used for the Homomorphic Classification.

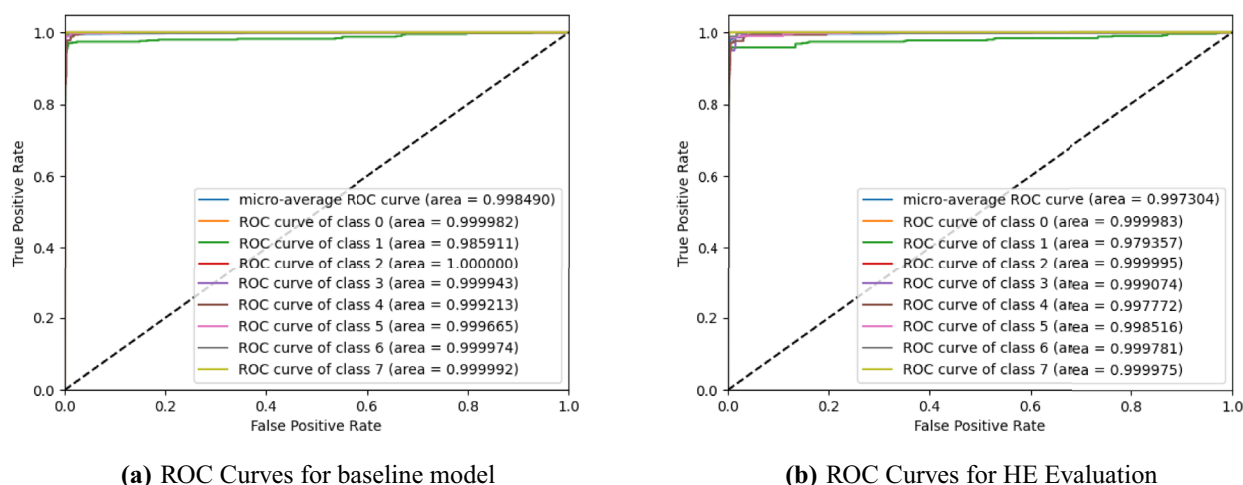
We report the ROC curves for the baseline model with and without HE in Fig. 2. The various time taken for different processes during the encrypted inference is reported in Table 4 and the amount of storage used during the encrypted classification is reported in Table 5. The short run-time and low storage indicate that there are little to no trade-offs to switching to a HE-based model.

Setting	Distribution of samples	# Variants per party	Average accuracy	Average AUROC
Centralized	—	—	0.986	0.992
Federated	Balanced	1	0.873	0.978
		2	0.946	0.995
		4	0.980	0.999
		8	0.984	0.998
	Imbalanced	2-4*	0.944	0.999
		7-8 <sup>#</sup>	0.975	0.998

**Table 2.** Model performance for centralized and federated settings. (\*) refers to a split configuration based on geographical locations across 6 parties and (<sup>#</sup>) denotes a random split of samples across 8 parties. In the federated setting, there is an increase in average model performance with greater variability of variants that each party holds.

# Variants per Party	$\epsilon = 0.1$		$\epsilon = 1.0$		$\epsilon = 3.0$		$\epsilon = \infty$ (no DP)	
	Average accuracy	Average AUROC	Average accuracy	Average AUROC	Average accuracy	Average AUROC	Average accuracy	Average AUROC
1	0.716	0.934	0.876	0.985	0.878	0.986	0.873	0.978
2	0.776	0.951	0.943	0.996	0.950	0.996	0.946	0.995
4	0.827	0.966	0.971	0.998	0.972	0.999	0.980	0.999
8	0.691	0.920	0.952	0.994	0.952	0.995	0.984	0.998
2-4*	0.338	0.710	0.938	0.985	0.982	0.997	0.994	0.999
7-8†	0.802	0.958	0.959	0.997	0.963	0.997	0.975	0.998

**Table 3.** Model performance for federated models with varying  $\epsilon$ . (\*) refers to a split configuration based on geographical locations across 6 parties and (†) denotes a random split of samples across 8 parties. We observe that model performance generally decreases with the addition of differential privacy and that lowering the privacy budget leads to models with poor performance. Nonetheless, at a reasonable privacy level of  $\epsilon = 1$ , there is little to no degradation of the model.



**Figure 2.** ROC curves for centralized model evaluation with and without HE. There is small to no loss in accuracy when evaluating the model homomorphically.

Process	Time taken (s)
Encoding model	0.128
Encrypting 4000 samples	0.044
Homomorphic inference	0.103
Decryption and decoding	0.023
Total time taken	0.298

**Table 4.** Time taken for homomorphic inference.

### Discussion

In this work, we have demonstrated how a machine learning model can be trained jointly and securely from several data sources with federated learning and differential privacy, and how inference on new samples can be achieved in a privacy-preserving manner via homomorphic encryption techniques. Data from each owner stays locally on-premise and is never exposed to other owners in the system throughout the whole process from model training to inference. Each data owner will not be able to learn anything about the data from other owners, beyond what can be inferred via the global model. Choosing an appropriate machine learning algorithm for the global model, for instance, logistic regression, will then restrict the sharing of information and prevent the exposure of individual data values. In contrast, machine learning models such as K-nearest neighbour are inappropriate as they will expose all the individual data values. We base our discussion here on a logistic regression

Object	Storage per object (kB)	Total storage (MB)
Secret key	385	0.385
Public key	193	0.193
Evaluation key	1200	1.2
Ciphertext	385	5.7
Plaintext	193	25
Total storage		32.5

**Table 5.** Storage consumption of homomorphic evaluation. There is a total of 128 Plaintext objects and 15 Ciphertext objects. See “Methods” for details on deriving the number of Plaintext and Ciphertext objects.

model. We note that our model can perform inference tasks directly on encrypted inputs, for instance, on new samples to be classified.

**Comparisons to related work.** Several recent developments in the cybersecurity domain have focused on training a model securely with technologies such as homomorphic encryption (HE)<sup>8–10</sup> and multi-party computation (MPC)<sup>11–13</sup>. Two recent works that presented frameworks using a combination of at least two privacy-preserving technologies have been published by Kaissis et al.<sup>14</sup> and Carpov et al.<sup>15</sup>.

Specifically, Kaissis et al. introduced a framework called PriMIA (Privacy-preserving Medical Image Analysis), which allows data owners to collaborate and train a medical image classification model securely via FL, utilizing DP which provides an additional layer of privacy. The evaluation of the model is then executed via a 2-party protocol<sup>16</sup> based on a type of MPC known as Function Secret Sharing, split between 2 servers. This means that, from a security aspect, there is a need to protect the 2 servers from malicious clients. CoVnita, on the other hand, only requires standard cryptographic key management, which is simpler to protect. In addition, the replacing of MPC with HE means that the evaluation phase of our solution does not require a pool of correlated randomness for effective use, and also can be extended to use 2-key HE with techniques described in Chen et al.<sup>17</sup>.

Carpov et al. argue that MPC is a better alternative to FL, as the latter may lead to possible leakage of information about the model during gradient updates. Their proposed framework (GenoPPML) utilizes both MPC and HE, where a logistic regression model is trained with MPC over 2 servers with a differentially private mechanism. New samples to be evaluated are then encrypted with HE before being classified by the model. In our work, we mitigate the information leakage concerns of Carpov et al. regarding federated learning by utilizing differential privacy, thus avoiding the hefty overheads of using MPC for more than 3 parties.

**Enabling privacy-preserving technologies via domain-aware data preprocessing.** Despite rapid developments in the cybersecurity space, many of the tools are not optimized to handle “-omics” data, which have far higher dimensionality than data from traditional fields such as image classification. Although GenoPPML<sup>15</sup> demonstrated feasibility on gene expression data, we note that the largest processed feature space in that work was 25,128. In comparison, the genomic sequence length of the SARS-CoV-2 viral strain here is approximately 30 kB, which would result in more than 90,000 processed features under a standard one-hot encoding schema, assuming the simplest case of limiting encoding to the four nucleotide bases.

It is useful to leverage domain knowledge to make such “-omics”-problems more tractable for privacy-preserving technologies. Good data preprocessing step can help reduce the dimensionality of raw data significantly while retaining important information relevant to the classification task. For instance, our work here has utilized Dashing, a hashing technique commonly used in genome classification, to provide a layer of abstraction from the raw sequence data. Further, we leverage biological knowledge that mutations in the S gene (which encodes the spike protein that influences infectivity), are key drivers of biological differences between the strains<sup>18</sup> to reduce the initial size of the raw data. Specifically, we truncated the first 20 kB of the genome sequence (the regions preceding the S gene) before Dashing and encryption. This allowed us to achieve faster data preprocessing and model training speeds and was used in our submission to the iDASH 2021 competition<sup>4</sup> that won first place.

**Limitations and future work.** While an honest-but-curious threat model is usually sufficient in most situations, we acknowledge that our framework is unable to defend against truly malicious adversaries. A malicious data provider could for instance contribute substandard data that would affect the quality of the trained model. We emphasize that while our current setup does not prevent such acts of vandalism that lead to the degradation of overall model performance, the privacy of data from each owner in the system will not be compromised by this.

Our proposed framework also does not provide proof to an end user that the computation had been performed correctly. For instance, issues arising during deployment may lead to a malfunctioning classifier, but an end user who submits a new sample will be none-the-wiser about this. This is a limitation in HE setups, as visibility on the ‘proof-of-computation’ is low or non-existent to the end user.

Although we have simulated a distributed set of data owners in this work, we note that there also exist centralized resources for SARS-CoV-2, such as GISAID’s EpiCoV platform, that serve as a trusted platform for researchers to share information. Our future work will consider how our technology can add value to such systems that

are based on a trusted central platform. Another future work would be to extend this framework to support other models or statistics (e.g. Kaplan–Meier survival analysis) and other forms of medical data (e.g. images).

## Methods

A figure outlining the whole process from data processing to encrypted inference is described in Fig. 3. Here, we provide preliminaries for the key technologies used in our framework and details of each component in the workflow above.

**Federated learning.** Federated learning (FL) is a technique in machine learning that allows multiple nodes to train models without exchanging data directly. It was originally developed by Google to train a global model across mobile devices using a central server<sup>19</sup>. Each node possesses a dataset on which they will locally process and provide an update to the global model. More precisely, each node train a local model  $w$ , over  $n$  samples, with the following objective function  $\min \frac{1}{n} \sum_{i=1}^n f_i(w)$ , where  $f_i(w)$  is usually set to be the loss between the prediction of the  $i$ -th sample and its actual value. Each node  $k$  locally computes  $g_k = \nabla f_i(w)$ . The central server then computes the aggregate of  $g_k$  and updates the model, for some fixed learning rate  $\eta$ , with  $w_{t+1} \leftarrow w_t + \eta \sum_k g_k$ . In this manner, the raw data is not shared between the nodes or the central server.

Based on the distribution of the data attributes and sample spaces, FL can be categorized into two broad categories—horizontal FL and vertical FL. Horizontal FL refers to each node having similar data attributes, but different sample spaces, while the nodes in vertical FL have different (and often unique) data attributes of the same set of samples. Horizontal FL can be further subdivided into two categories based on the number of data points each node possesses. If all nodes have an identical number of data points, then it is labelled as an independent, identically distributed (IID) distribution and otherwise, a non-IID distribution.

**Homomorphic encryption.** Homomorphic encryption (HE) is a special type of encryption scheme that allows computation to be performed on encrypted data. It was first proposed by<sup>20</sup>, with the first construction achieved by<sup>21</sup>. HE schemes are “noisy” in general, where noise is applied to a message as part of the encryption process to mask its value. A noise budget is set when the encryption scheme is initialized and computation (called homomorphic operations) on encrypted data consumes it. Once the noise budget is fully depleted, decrypting the ciphertext would result in an incorrect result.

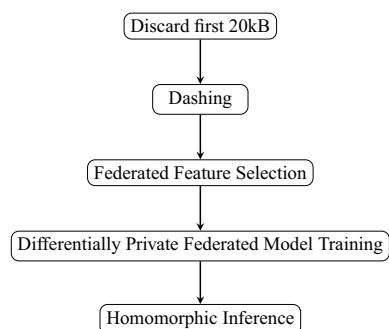
In this work, we use the CKKS scheme<sup>22</sup> which supports homomorphic operations on encrypted approximate numbers. Each number is encrypted with an initial precision and computation gradually reduces it. Thus, the decrypted message is an approximation of the true computation result.

Crucial to practical performance, HE schemes can store and simultaneously operate on more data in a single ciphertext by leveraging the decomposition of the plaintext space  $R = \mathbb{Z}[x]/(x^N + 1)$ <sup>23</sup>. The CKKS scheme supports the encoding of  $\frac{N}{2}$  complex numbers into a degree  $N - 1$  polynomial via the canonical map  $\phi : \mathbb{C}^{N/2} \rightarrow R$ . This process is described by the following functions:

- $m(x) = \text{Encode}(z_0, z_1, \dots, z_{N/2-1}) = \phi(z_0, z_1, \dots, z_{N/2-1})$ .
- $(z_0, z_1, \dots, z_{N/2-1}) = \text{Decode}(m(x)) = \phi^{-1}(m(x))$ .

Each number is encoded into a slot of the ciphertext ( $N/2$  many in each). This reduces the number of ciphertexts required in applications and each homomorphic operation is done on all slots in parallel, i.e. adding and multiplying ciphertexts result in the same operation applied to all slots respectively. There is also an inter-slot data movement mechanism that will return a ciphertext whose slots are rotations of those in its input.

**Differentially private stochastic gradient descent.** Differential privacy (DP) is a privacy mechanism that protects an individual’s data when it is used in a database. A formal definition proposed by<sup>24</sup> states that



**Figure 3.** CoVnita workflow. The first 20 kB is first discarded to reduce the size of the data. A tool called Dashing is used to transform the truncated genomic sequence into 512 features, each a 64-bit hash value. The parties perform a federated feature selection to further reduce the number of features to 15. A model is then trained in jointly with FL and DP. The test samples are encrypted and evaluated with HE.

for two datasets  $D$  and  $D'$  differing in at most one record, given an algorithm  $\mathcal{M}$ , we say that  $\mathcal{M}$  is  $(\epsilon, \delta)$ -differentially private if  $\text{IP}[\mathcal{M}(D) = x] \leq \exp(\epsilon) \cdot \text{IP}[\mathcal{M}(D') = x] + \delta$ . The parameter  $\epsilon$  can be thought of as the privacy budget or the largest distance between the outputs of  $\mathcal{M}$  on the datasets. If  $\epsilon = 0$ , it is equivalent to having different datasets giving the same output. The parameter  $\delta$  on the other hand represents the probability of the individual's data leaking. Differentially private mechanisms have an interesting property of being robust to post-processing. This means that any function applied to the output of any differentially private mechanism is also differentially private.

Stochastic gradient descent (SGD) is an iterative method commonly used in machine learning algorithms. It is used as a low-cost alternative to other second-order methods for finding the local minimum of the objective function, at the expense of a lower convergence rate.

In machine learning, the de facto standard would be to add DP during the model training, specifically to stochastic gradient descent (SGD). This allows the model to be distributed subsequently while ensuring the privacy of the data used for training. Abadi et al.<sup>25</sup> proposed the following method of applying DP to SGD; First, compute the gradients of the loss function (for each feature). Next, clip the gradients such that the gradient vector has a norm less than some predetermined threshold. Finally, add a suitable amount of Gaussian noise.

**Data and preprocessing.** We selected eight COVID-19 strains namely *B.1.1.7* (Alpha), *B.1.429* (Epsilon), *P.1* (Gamma), *B.1.526* (Iota), *B.1.617.2* (Delta), *C.37* (Lambda), *B.1.621* (Mu) and *B.1.1.529* (Omicron). For each strain, we obtained 2500 sequences, of which 500 were set aside as a held-out test set for evaluating model performance. The samples for the *B.1.1.7* (Alpha), *B.1.429* (Epsilon), *P.1* (Gamma) and *B.1.526* (Iota) strains were the same ones provided in the iDASH 2021 competition<sup>4</sup>. The remaining samples for the *B.1.617.2* (Delta), *C.37* (Lambda), *B.1.621* (Mu) and *B.1.1.529* (Omicron) strains were obtained from the Global Initiative on Sharing Avian Influenza Data (GISAID) database<sup>26–28</sup> (accessed on 31 Dec 2021).

As viral strains are typically defined based on their phenotypic characteristics rather than simple sequence similarity<sup>29,30</sup>, alignment-free methods<sup>31,32</sup> are better suited to perform the classification. These methods transform raw genomic sequences into feature vectors that are then used to train machine learning models.

Dashing<sup>33</sup> is a tool used to estimate the similarities of two genomic sequences. For each genomic sequence, we truncate the first 20 kB and then split the remaining into  $k$ -mers, where we chose  $k = 31$ , as tested in<sup>33</sup>. Each  $k$ -mer is then converted into a 64-bit hash. The similarities of the genomic sequences (or equivalently, approximate distance) can then be computed by checking if a hash value of one of the sequences appeared in the other. The HyperLogLog sketch<sup>34</sup> is used to estimate the cardinality of the resulting hash sets. More precisely, the hash values are sorted into buckets via a predetermined prefix and the sketch value is given as the maximum leading zero count. We chose to set the length of the prefix to be 9, giving us a total of 512 buckets, or equivalently, features representing each genomic sequence.

Although Dashing can provide some form of privacy as it transforms raw genomic sequences into an abstract hash value, the process is not irreversible and thus not privacy-preserving.

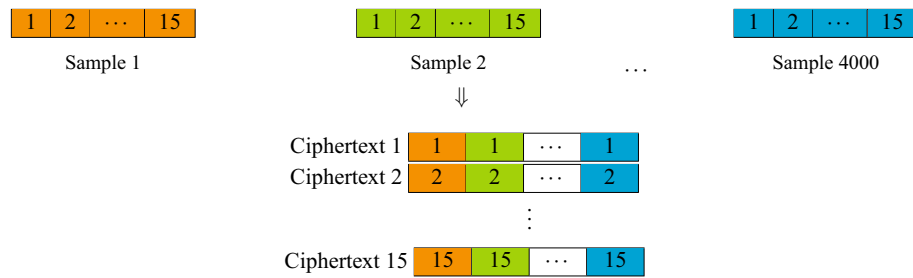
**Differentially private federated model training.** Due to a large amount of genomic data, we must select a sufficiently small subset of features that contains the most genomic information, to make the model training process tractable. Differential privacy is deployed during the training of local models before these models are combined into a global model.

*Federated feature selection.* The  $\chi^2$ -test is a popular correlation test used to test the correlation between a feature and the response. A larger  $\chi^2$  value indicates that the feature and the response suggest a higher correlation and should be selected for the training of the model. In a traditional machine learning setting, feature selection is performed with the expectation that all the required data reside in the same machine. However, in this work, the data is split across several parties and cannot be directly shared, or pooled amongst the parties for feature selection. Thus a federated version of feature selection is necessary. We implemented a federated version of the  $\chi^2$ -test proposed by Wang et al. in<sup>5</sup>. They proposed that in the federated setting, the  $\chi^2$ -test can be approximated by its 2nd frequency moments. Based on our empirical testing, we find that a selection of 15 features is optimal.

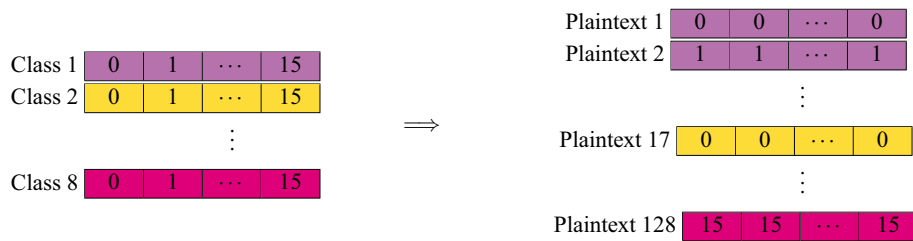
*Differentially private federated learning.* We represented the sketch from Dashing as a one-hot vector and trained a logistic regression model using a differentially private SGD provided by Opacus. Opacus<sup>35</sup> is a library that supports DP with PyTorch<sup>36</sup>. We used the cross entropy loss function with a learning rate of 0.01 and a default 60 training epochs. The value of  $\epsilon$  was varied and  $\delta$  was set to  $\frac{1}{D}$ , where  $D$  is the total number of samples used to train the model.

**Homomorphic inference.** The evaluation of the logistic regression model can be viewed as applying the sigmoid function on the inner product between the model weights and the features of the evaluating sample. As the sigmoid function serves to map the inner product output to probabilistic outcomes, we omit to evaluate the sigmoid function in the encrypted domain and instead determine the predicted class by choosing the largest value. We leverage the ability to store and operate on multiple data within a HE ciphertext to evaluate multiple samples simultaneously. The packing method we use in this implementation is based on<sup>37,38</sup>. The main idea would be to pack one feature from each sample into a single ciphertext. One plaintext-ciphertext multiplication is then performed and the resulting ciphertext is summed together to obtain the evaluation of the model on the new samples.

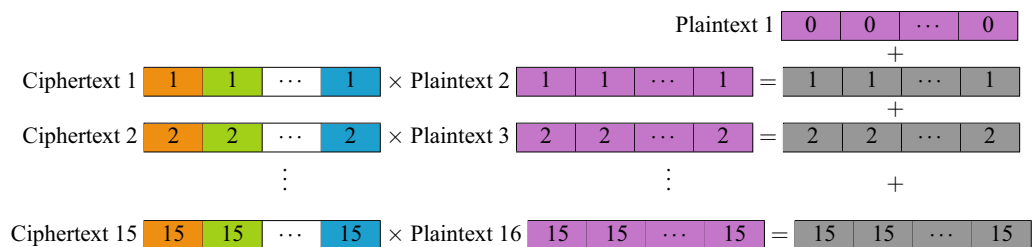
More precisely, the homomorphic inference first begins by Dashing the new samples, converting each sample into 64-bit hashes and the 15 chosen features are selected. The encryption process is shown in Fig. 4. The model, on the other hand, is encoded as depicted in Fig. 5. The homomorphic classification based on such an encoding method require Plaintext–Ciphertext multiplications and Ciphertext additions, described in Fig. 6.



**Figure 4.** Encrypting samples. The same feature from all samples is encrypted into a single ciphertext. Since there are 15 features, a total of 15 ciphertext is used.



**Figure 5.** Encoding the model. Each feature is encoded multiple times in a single Plaintext object. A total of  $8 \times 16 = 128$  Plaintext objects (15 features and 1 bias) is used.



**Figure 6.** Homomorphic inference for a class. To evaluate if a sample belongs to a class, 15 plaintext–ciphertext multiplications are performed, followed by 15 summations of the resultant ciphertext and the bias Plaintext object. This allows all 4000 samples to be evaluated for a class simultaneously.



## Data availability

The data for the following strains-B.1.1.7 (Alpha), B.1.429 (Epsilon), P.1 (Gamma) and B.1.526 (Iota) are available from the organizers of the iDASH'2021 competition at <http://www.humangenomeprivacy.org/2021/contact.html>, which were used under license for the current study, and so are not publicly available. Data are however available from the corresponding author, Jun Jie Sim, upon reasonable request and with permission of the organizers of the iDASH'2021 competition. The data for the remaining strains - B.1.617.2 (Delta), C.37 (Lambda), B.1.621 (Mu) and B.1.1.529 (Omicron) are available in the GISAID repository, with Episet ID: EPI\_SET\_220924cw at <https://doi.org/10.55876/gis8.220924cw>.

Received: 16 October 2022; Accepted: 3 May 2023

Published online: 08 May 2023

## References

1. Cha, S. A little known cult is at the heart of S. Korea's latest covid-19 outbreak. <https://www.reuters.com/> (2021).
2. Chen, L. Singapore sees most covid-19 cases in 10 months after karaoke cluster. <https://www.reuters.com/> (2021).
3. Kuo, T.-T. *et al.* The evolving privacy and security concerns for genomic data analysis and sharing as observed from the iDASH competition. *J. Am. Med. Inf. Assoc.* <https://doi.org/10.1093/jamia/ocac165> (2022).
4. iDASH Privacy & Security Workshop 2021 Secure Genome Analysis Competition. Track ii: Homomorphic encryption-based secure viral strain classification (2021).
5. Wang, L., Pang, Q., Wang, S. & Song, D. Fed- $\chi_2$ : Privacy preserving federated correlation test. [arXiv:abs/2105.14618](https://arxiv.org/abs/2105.14618) (CoRR) (2021).
6. Microsoft SEAL (release 3.2.2). Microsoft Research, Redmond, WA (2019). <https://github.com/Microsoft/SEAL>.
7. Albrecht, M. R., Player, R. & Scott, S. lwe-estimator commit c50ab18. <https://github.com/malb/lattice-estimator/> (2022).
8. Gilad-Bachrach, R. *et al.* Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy. In *Proceedings of The 33rd International Conference on Machine Learning*, vol. 48 of *Proceedings of Machine Learning Research* Balcan, M. F. & Weinberger, K. Q. (eds.), 201–210 (PMLR, 2016).
9. Chou, E. *et al.* Faster cryptonets: Leveraging sparsity for real-world encrypted inference. <https://doi.org/10.48550/ARXIV.1811.09953> (2018).
10. Froelicher, D. *et al.* Truly privacy-preserving federated analytics for precision medicine with multiparty homomorphic encryption. *BioRxiv* <https://doi.org/10.1101/2021.02.24.432489> (2021).
11. Mohassel, P. & Zhang, Y. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE Symposium on Security and Privacy (SP)*, 19–38. <https://doi.org/10.1109/SP.2017.12> (2017).
12. Wagh, S., Gupta, D. & Chandran, N. Securenn: Efficient and private neural network training. *Cryptology ePrint Archive*, Paper 2018/442 (2018). <https://eprint.iacr.org/2018/442>.
13. Wagh, S. *et al.* Falcon: Honest-majority maliciously secure framework for private deep learning. <https://doi.org/10.48550/ARXIV.2004.02229> (2020).
14. Kaissis, G. *et al.* End-to-end privacy preserving deep learning on multi-institutional medical imaging. *Nat. Mach. Intell.* **3**, 1–12. <https://doi.org/10.1038/s42256-021-00337-8> (2021).
15. Carпов, S., Gama, N., Georgieva, M. & Jetchev, D. Genoppml—a framework for genomic privacy-preserving machine learning. *Cryptology ePrint Archive*, Paper 2021/733 (2021). <https://eprint.iacr.org/2021/733>.
16. Ryffel, T., Pointcheval, D. & Bach, F. R. ARIANN: Low-interaction privacy-preserving deep learning via function secret sharing. [arXiv:abs/2006.04593](https://arxiv.org/abs/2006.04593) (CoRR) (2020).
17. Chen, H., Dai, W., Kim, M. & Song, Y. Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference. *Cryptology ePrint Archive*, Paper 2019/524. <https://doi.org/10.1145/3319535.3363207> (2019). <https://eprint.iacr.org/2019/524>.
18. Harvey, W. T. *et al.* Sars-cov-2 variants, spike mutations and immune escape. *Nat. Rev. Microbiol.* **19**, 409–424 (2021).
19. McMahan, H. B., Moore, E., Ramage, D. & Arcas, B. A. Federated learning of deep networks using model averaging. [arXiv:abs/1602.05629](https://arxiv.org/abs/1602.05629) (CoRR) (2016).
20. Rivest, R. L., Adleman, L. & Dertouzos, M. L. *On Data Banks and Privacy Homomorphisms. Foundations of Secure Computation* (Academia Press, 1978).
21. Gentry, C. Fully homomorphic encryption using ideal lattices. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, 169–178 (Association for Computing Machinery, 2009). <https://doi.org/10.1145/1536414.1536440>.
22. Cheon, J. H., Kim, A., Kim, M. & Song, Y. Homomorphic encryption for arithmetic of approximate numbers. *Cryptology ePrint Archive*, Paper 2016/421 (2016). <https://eprint.iacr.org/2016/421>.
23. Smart, N. P. & Vercauteren, F. Fully homomorphic simd operations. *Cryptology ePrint Archive*, Paper 2011/133 (2011). <https://eprint.iacr.org/2011/133>.
24. Dwork, C., McSherry, F., Nissim, K. & Smith, A. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography* (eds Halevi, S. & Rabin, T.) 265–284 (Springer, 2006).
25. Abadi, M. *et al.* Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. <https://doi.org/10.1145/2976749.2978318> (ACM, 2016).
26. Khare, S. *et al.* Gisaids's role in pandemic response. *China CDC Weekly* **3**, 1049. <https://doi.org/10.46234/ccdcw2021.255> (2021).
27. Elbe, S. & Buckland Merrett, G. Data, disease and diplomacy: Gisaids's innovative contribution to global health: Data, disease and diplomacy. *Glob. Challenges* **1**, 33–46. <https://doi.org/10.1002/gch2.1018> (2017).
28. Shu, Y. & McCauley, J. Gisaids: Global initiative on sharing all influenza data—from vision to reality. *Eurosurveillance* <https://doi.org/10.2807/1560-7917.ES.2017.22.13.30494> (2017).
29. Kuhn, J. H. *et al.* Virus nomenclature below the species level: A standardized nomenclature for laboratory animal-adapted strains and variants of viruses assigned to the family filoviridae. *Adv. Virol.* **158**, 1425–1432 (2013).
30. Mascola, J. R., Graham, B. S. & Fauci, A. S. Sars-cov-2 viral variants-tackling a moving target. *JAMA* **325**, 1261–1262 (2021).
31. Xing, Z., Pei, J. & Keogh, E. A brief survey on sequence classification. *ACM SIGKDD Explor. News* **12**, 40–48 (2010).
32. Vinga, S. & Almeida, J. Alignment-free sequence comparison—a review. *Bioinformatics* **19**, 513–523 (2003).
33. Baker, D. N. & Langmead, B. Dashing: Fast and accurate genomic distances with HyperLogLog. *Genome Biol.* **20**, 265 (2019).
34. Flajolet, P., Fusy, É., Gandouet, O. & Meunier, F. HyperLogLog: the analysis of a near-optimal cardinality estimation algorithm. In Jacquet, P. (ed.) *AofA: Analysis of Algorithms*, vol. DMTCS Proceedings vol. AH, 2007 Conference on Analysis of Algorithms (AofA 07) of *DMTCS Proceedings*, 137–156 (Discrete Mathematics and Theoretical Computer Science, 2007). <https://doi.org/10.46298/dmtcs.3545>.
35. Yousefpour, A. *et al.* Opacus: User-friendly differential privacy library in PyTorch. [arXiv:2109.12298](https://arxiv.org/abs/2109.12298) (arXiv preprint) (2021).
36. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* Vol. 32 (eds Wallach, H. *et al.*) 8024–8035 (Curran Associates Inc., 2019).

37. Al Badawi, A. *et al.* Towards the alexnet moment for homomorphic encryption: Hcnn, the first homomorphic cnn on encrypted data with gpus. *IEEE Trans. Emerg. Top. Comput.* **9**, 1330–1343. <https://doi.org/10.1109/TETC.2020.3014636> (2021).
38. Chan, F. M. *et al.* Genotype imputation with homomorphic encryption. In *2021 6th International Conference on Biomedical Signal and Image Processing, ICBIP '21*, 9–13 (Association for Computing Machinery, 2021). <https://doi.org/10.1145/3484424.3484426>.

### Acknowledgements

The authors would like to thank Dr Sebastian Maurer-Stroh for his assistance in obtaining the GISAID EpiSet ID and the iDASH 2021 organizers for the original problem statement of secure viral classification. This research is supported by Institute for Infocomm Research, A\*STAR Research Entities under its RIE2020 Advanced Manufacturing and Engineering (AME) Programmatic Program (Award A19E3b0099).

### Author contributions

J.J.S.: conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing, visualization, project administration. W.Z.: conceptualization, methodology, validation, formal analysis, resources, writing. F.M.C.: methodology, software, investigation, data curation. M.S.M.S.A.: Methodology, software, investigation. X.D.: software, investigation, resources. B.H.M.T.: conceptualization, methodology, formal analysis, writing, supervision. K.M.M.A.: conceptualization, supervision, project administration, funding acquisition. All authors read and approved the final manuscript.

### Competing interests

The authors declare no competing interests.

### Additional information

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1038/s41598-023-34535-8>.

**Correspondence** and requests for materials should be addressed to J.J.S.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

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



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023