# Privacy-Preserving Artificial Intelligence: Application to Precision Medicine

Anamaria Vizitiu[1,2], Cosmin Ioan Niță[1,2], Andrei Puiu[1,2], Constantin Suciu[1,2] and Lucian Mihai Itu[1,2]

***Abstract*— Motivated by state-of-the-art performances across a wide variety of areas, over the last few years Machine Learning has drawn a significant amount of attention from the healthcare domain. Despite their potential in enabling personalized medicine applications, the adoption of Deep Learning based solutions in clinical workflows has been hindered in many cases by the strict regulations concerning the privacy of patient health data. We propose a solution that relies on Fully Homomorphic Encryption, particularly on the MORE scheme, as a mechanism for enabling computations on sensitive health data, without revealing the underlying data. The chosen variant of the encryption scheme allows for the computations in the Neural Network model to be directly performed on floating point numbers, while incurring a reasonably small computational overhead. For feasibility evaluation, we demonstrate on the MNIST digit recognition task that Deep Learning can be performed on encrypted data without compromising the accuracy. We then address a more complex task by training a model on encrypted data to estimate the outputs of a whole-body circulation (WBC) model. These results underline the potential of the proposed approach to outperform current solutions by delivering comparable results to the unencrypted Deep Learning based solutions, in a reasonable amount of time. Lastly, the security aspects of the encryption scheme are analyzed, and we show that, even though the chosen encryption scheme favors performance and utility at the cost of weaker security, it can still be used in certain practical applications.**

## I. Introduction

In recent years machine learning algorithms, and specifically Deep Neural Networks, have shown promising results in delivering personalized medicine, allowing for tailored diagnosis, treatment planning and disease prevention [1]. Since Deep Neural Networks have the ability to learn from past observations, they represent an attractive solution for integrating the knowledge and experience of medical experts into Computer-aided Detection (CADe) solutions.

Machine learning relies extensively on existing and future patient data to deliver accurate and reliable results [2]. However, among all types of data associated with an individual, medical data has some of the highest privacy requirements. Thus, as access to sensitive plaintext data is required in deep learning based applications, privacy and security concerns have been raised [3]. Moreover, the currently adopted regulations towards confidentiality guarantees for personal data manipulation (e.g. GDPR in EU, HIPAA in USA) urges for the adoption of more effective privacy-preserving techniques.

Typically, to export sensitive data without compromising privacy, proper anonymization has to be performed. Thus, some of the data properties are modified, leading to a trade-off between privacy and utility. To address this limitation, herein we rely on a specific form of encryption, called homomorphic encryption, which represents a promising solution for guaranteeing privacy while still maintaining full utility. Specifically, the chosen homomorphic encryption scheme (MORE) [4] enables a limited set of operations to be performed directly on encrypted data, without having to reveal the underlying data or the encryption key. This ensures that both data and predictions remain private and data is analyzed in its encrypted from.

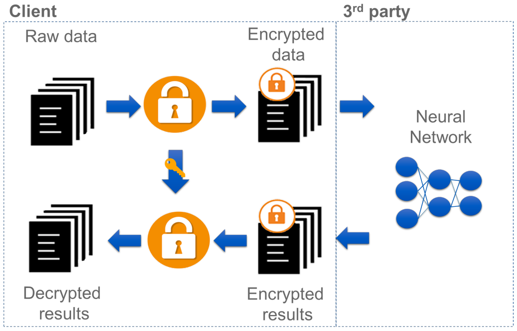

Fig. 1. Workflow of a privacy-preserving deep learning based application relying on homomorphic encryption.

This property is particularly useful in the context of deep learning solutions. As outlined in Figure 1, privacy is preserved at three levels: (i) during training, when the external party (e.g. a cloud or processor) processes directly ciphertexts, (ii) during inference, when the patient's data remains confidential: algorithm receives as input ciphertexts and outputs ciphertexts, which are revealed only on the client side after performing the decryption, and (iii) the external party's deep learning model remains confidential. Consequently, the secure processing of medical data is performed in such a way that the external party cannot derive knowledge from the data, and the user is unable to obtain information regarding the machine learning model.

Driven by the difficulties that arise in practice when employing deep learning over encrypted data and also by the inefficiency of current solutions, herein we propose a method that increases the efficiency of the encrypted models in real-world applications by enabling: (i) computations over rational numbers, (ii) faster operations and, (iii) results comparable to those obtained with the unencrypted model. We assess the feasibility of the proposed solution for delivering reliable results, and show that performance is not lost when

[1] Department of Automation and Information Technology, Transilvania University of Brașov, Brașov, Romania

[2] Corporate Technology, Siemens SRL, Brașov, Romania

deep neural networks operate on data encrypted using the MORE homomorphic encryption scheme. We evaluate the privacy-preserving deep learning algorithms on the classic benchmarking application of digit classification, and on a personalized medicine application.

## II. Related work

Recent advances in homomorphic encryption have lead to several encryption schemes, with different properties and constraints. The most notable drawback of the majority of fully homomorphic encryption (FHE) schemes is that each operation adds noise to the underlying message, therefore limiting the overall number of operations that can be performed without losing too much accuracy. Furthermore, to the best of our knowledge, there is no currently available partially or fully homomorphic encryption scheme that can process rational numbers (only integer numbers are supported). As a consequence, a variant of a matrix-based method, called MORE (Matrix Operation for Randomization or Encryption) [4] was adapted in the current work. Compared to currently studied schemes, in the context of privacy-preserving networks [5], [6], [7], MORE is noise free (unlimited number of operations can be performed on ciphertext data) and nondeterministic (multiple encryptions of the same message and with the same key result in different ciphertexts). Moreover, both division and multiplication operations can be performed over encrypted data.

While fully homomorphic encryption seems to offer a high level solution for privacy-preserving computations with deep learning models, there are still important practical challenges that urge for stronger security, faster running time, and improved generalization performance [8].

To empower privacy-preserving computations in the context of deep learning, it is crucial for the encryption scheme to be applicable to rational numbers. Previously reported approaches for handling this aspect rely on the encoding of rational numbers as integers or as a sequence of integers [9]. Such an approach has limited usability since it does not allow for any operation to be performed on the encoded form. Moreover, adopting an encoding strategy as a way of enabling computations to be performed on real-data introduces not only a clear limitation in its utility but also directly affects the outcome of the computations. To address this limitation, the MORE encryption scheme was adapted to directly support floating point arithmetic. A more detailed description is provided in section III-A.

## III. Matrix based data randomization

With Gentry's first introduction of a fully homomorphic encryption schemes [10], numerous variations of the original strategy were proposed in literature [11]. While most of the schemes were shown to be secure, they suffer from very poor performance, being several orders of magnitude slower than the plaintext computations. Alternatively to the original fully homomorphic encryption schemes, some simpler methods which are based on linear transformations emerged. Although criticized due to weaker security [12], [13], this class of methods appears to be currently the only practical approach for performing privacy-preserving computations in real-world applications.

Herein we have employed a variant of the MORE encryption scheme. The MORE scheme relies on matrix algebra and can be used to encrypt a numerical value as a matrix. Therefore, operations performed on encrypted values will turn into matrix operations, e.g. addition of unencrypted scalars will result in addition of encrypted matrices. The MORE encryption scheme is defined as follows. It can be directly generalized to $n$ by $n$ matrices, however, for simplicity, herein we present only the 2 by 2 setup:

1) Let $m$ be the scalar value to be encrypted
2) Let $S$ be a 2 by 2 invertible matrix, representing the encryption and decryption key
3) $m$ is mapped to a 2 by 2 matrix $M$ as follows: $M = \begin{pmatrix} m & 0 \\ 0 & r \end{pmatrix}$ where $r$ is a random parameter
4) Encryption: $C = SMS^{-1}$, $C$ is the encrypted matrix
5) Decryption $M = S^{-1}MS$, the element on the first row and column represent the plaintext value

The MORE scheme is fully homomorphic with respect to algebraic operations, i.e. given two encrypted matrices $C_1 = SM_1S^{-1}$ and $C_2 = SM_2S^{-1}$, for multiplication $C_1C_2 = SM_1S^{-1}SM_2S^{-1} = SM_1M_2S^{-1}$, which is the encryption of the multiplication $M_1M_2$, and for addition $M_1 + M_2 = SM_1S^{-1} + SM_2S^{-1} = S(M_1 + M_2)S^{-1}$. Similarly, this applies also for subtraction and division, and even for operations involving unencrypted scalars.

### A. Encryption of rational numbers

The original MORE scheme, as described by Kipnis et al. [4], applies the encryption to positive integer numbers modulo $N$, and all the operations are performed modulo $N$. This is a typical characteristic of fully homomorphic or partially homomorphic encryption schemes. Typical approaches for extending the methodology to rational numbers consist in employing an encoding operation. More specifically rational numbers are first encoded as integers, or as a sequence of integer numbers, and then the encryption is applied on the resulting encoded form. In essence, this is a straightforward problem but, no solution has been found to date in the context of homomorphic encryption. One approach consists in encoding rational numbers as continued fractions [9], however it is difficult to perform operations on numbers represented under this form. Another approach consists in turning rational numbers into an integer by multiplying with a large scale factor. Unfortunately this approach will not allow for divisions as it will cause the large scale factor to be reduced.

One of the most important advantages of the MORE encryption scheme is that it can also be directly applied on rational numbers without the need of an encoding operation. The drawback is that the method becomes vulnerable to known ciphertext attacks, as described in Section V-C.

### B. Performing operations over encrypted data

It was shown previously that the MORE method is fully homomorphic with respect to algebraic operations. In real world applications, a broader spectrum of operations need to be performed, e.g. non-linear (exponential, logarithmic, square root, etc), comparison operations, etc. Typical approaches for performing non-linear operations consist in approximating the given functions as finite polynomial series (e.g. truncated Taylor series), therefore relying only on algebraic operations. The MORE scheme allows for a simple approach for performing such operations.

Knowing that operations performed on encrypted values turn into matrix operations, an intuitive approach is to compute most of the non-linear functions used in neural networks as matrix functions. However, a second approach can also be derived using a property of the MORE scheme: the secret message will always be one of the eigenvalues of the encrypted matrix, e.g. for the 2x2 case, the encrypted matrix $C$ will have two eigenvalues: $m$ and $r$ corresponding to the message and the random secret. If the random secret $r$ is chosen to be statistically indistinguishable from the message, it is impossible to separate the two without knowing the decryption matrix $S$. Therefore, given an encrypted matrix $C$, and knowing that $m$ is one of the eigenvalues of $C$, one can perform eigen decomposition, and then evaluate the given non-linear function directly on the eigenvalues of $C$. More specifically, given the eigen decomposition $VLV^{-1}$ where $V$ is the eigenvector matrix, and $L$ is the diagonal matrix containing the eigenvalues, one can evaluate any unary function by performing the evaluation separately for each eigenvalue $L_1, L_2, \ldots, L_n$ and then reconstructing the new encrypted matrix as $C_f = Vf(L)V^{-1}$. This approach can even be used for comparing an encrypted matrix $C$ with a plain scalar $s$. Non-linear binary operations involving two encrypted values cannot be performed, but these types of operations can be avoided in deep learning based applications.

## IV. Deep Neural Networks over Encrypted Data

In this section we evaluate the proposed encryption scheme in two types of deep learning applications: classification and regression. We first address a well known benchmarking application (digit classification), and then focus on training a neural network model on encrypted data to assess whole-body hemodynamics. Experiments demonstrate that we can ensure data security and, at the same time, efficiently perform deep learning based data analysis.

### A. MNIST: a typical dataset for neural networks

The MNIST (Modified National Institute of Standards and Technology database) dataset [14] contains images representing handwritten digits, and is ly used as reference for benchmarking image classification algorithms. The training dataset consists of 60,000 grayscale images, of relatively small dimension, 28x28, each image being labeled with the digit it depicts.

To address the challenge of privacy-preserving computations and evaluate the use of deep neural network models over encrypted data, the focus lies on solving the classification problem using a convolutional neural network (CNN) employed on encrypted input-output value pairs. Therefore, with a message $m \in \mathbb{R}$ encoded as a matrix $M \in \mathbb{R}^{2x2}$, for a training example, both the input image and the associated label vector are now represented as ciphertexts in the $\mathbb{R}^{28x28x2x2}$, and $\mathbb{R}^{10x2x2}$ domains. By leveraging the homomorphic property of the scheme, and with the direct support for floating-point arithmetic, training can be performed in a straightforward way.

The trained network has 6 layers, organized as follows: conv-pool-conv-pool-fc-fc. The first convolutional layer has 8 filters, the second one 16 filters, and both layers handle kernels of size 3x3. The pooling layer downsamples the images by a factor of two through averaging. The last two fully connected layers cover 100 and, respectively 10 nodes and all the activation functions employed in the network, except for the last layer, are sigmoid functions. The network was trained using stochastic gradient descent (SGD) to minimize a cross entropy loss between encrypted targets and encrypted predictions. A learning rate of 0.01 was considered, and training was performed in batches of 32 images for a number of 100 epochs. This network leads to an accuracy of 98.3% on the test dataset.

### B. Whole-body circulation model

*1) Introduction:* To demonstrate the feasibility of the proposed approach within a personalized medicine application, we have chosen a hemodynamic model of the cardiovascular system, i.e. a whole body circulation (WBC) model. Due to the prohibitive computational cost of spatial blood flow models (three-dimensional models in particular), closed loop models of the cardiovascular system rely heavily on lumped parameter modeling techniques, which are based on the analogy between hydraulics and electricity. The WBC model employed herein, displayed in Figure 2, contains a heart model (left ventricle (LV) and atrium, right ventricle and atrium, valves), the systemic circulation (arteries, capillaries, veins), and the pulmonary circulation (arteries, capillaries, veins) [15].

Time-varying elastance models are used for all four cham-

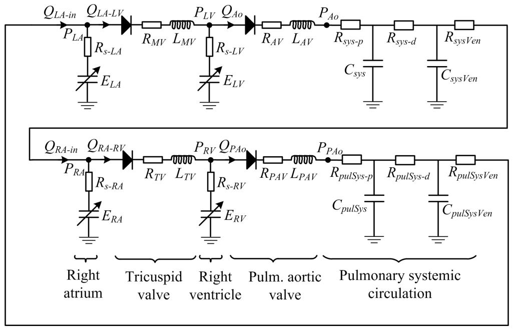

Fig. 2. Lumped parameter closed loop model of the cardiovascular system.

bers of the heart:

$$P(t) = E(t) \cdot (V(t) - V_0) - R_S Q(t), \qquad (1)$$

where $E$ is the time-varying elastance, $V$ is the cavity volume, $V_0$ is the dead volume of the cavity, and $R_s$ is a source resistance which accounts for the dependence between the flow and the cavity pressure [16] ($R_s = K_s E(t)(V(t) - V_0)$, $K_s$ – constant). The cavity volume is equal to:

$$\frac{dV}{dt} = Q_{in} - Q_{out}. \qquad (2)$$

The models of all four valves (mitral, aortic, tricuspid and pulmonary) of the heart include a resistance, an inertance and a diode (for simulating the opening and the closure of the valve based on the pressure gradient between the two sides of the valve). When the valve is closed, the flow across the valve is set to zero. When the valve is open, the following relationship holds:

$$P_{in} - P_{out} = R_{valve} \cdot Q + L_{valve} \cdot \frac{dV}{dt}, \qquad (3)$$

where $P_{in}$ and $P_{out}$ represent the pressures at the inlet and respectively the outlet of the valve. Each valve opens when $P_{in}$ becomes greater than $P_{out}$, and closes when the flow rate becomes negative. A three-element Windkessel model is used for the systemic circulation, represented by the following relationship between instantaneous flow and pressure:

$$\frac{dP_{Ao}}{dt} = R_{sys-p}\frac{dQ_{Ao}}{dt} - \frac{P_{Ao} - P_{ven}}{R_{sys-d} \cdot C_{sys}} + \frac{Q_{Ao}(R_{sys-p} - R_{sys-d})}{R_{sys-d} \cdot C_{sys}}, \qquad (4)$$

where $R_{sys-p}$ and $R_{sys-d}$ are the proximal and distal resistances respectively, $C_{sys}$ is the compliance, and $P_{ven}$ is the venous pressure. A two-element Windkessel model is used for the systemic venous circulation:

$$\frac{dP_{ven}}{dt} = \frac{Q_{ven}}{C_{sysVen}} - \frac{dP_{ven} - P_{RA}}{R_{sysVen} \cdot C_{sysVen}}. \qquad (5)$$

Similar models are employed for the pulmonary circulation.

*2) Personalization:* The above described WBC model may be run under patient-specific conditions to compute various clinically relevant measures of interest: arterial resistance, arterial compliance, dead volume of the left / right ventricle, stroke work, ventricular / atrial / arterial elastance, arterial ventricular coupling, pressure-volume loop, etc. Thus, model parameters need to be personalized to match the patient-specific conditions and state.

The personalization framework used herein has been previously described in detail [17], and consists of two sequential steps. First, a series of parameters are computed directly, and next, a fully automatic optimization-based calibration method is employed to estimate the values of the remaining parameters, ensuring that the personalized computations match the measurements.

The patient-specific input parameters are:

- Systemic circulation: peak aortic systolic pressure, end-diastolic aortic pressure, left ventricular end-systolic and end-diastolic volumes, left ventricular ejection time
- Pulmonary circulation: peak pulmonary artery systolic pressure, end-diastolic pulmonary artery pressure, right ventricular end-systolic and end-diastolic volumes, right ventricular ejection time

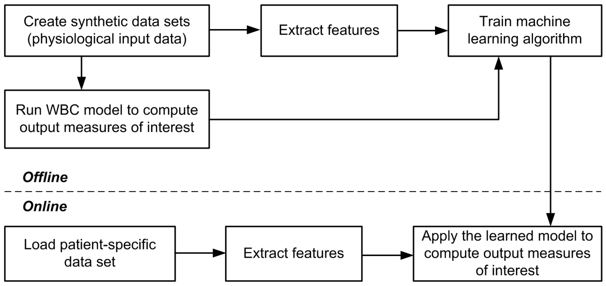
Create synthetic data sets (physiological input data)
Extract features
Train machine learning algorithm
Run WBC model to compute output measures of interest
Offline
Online
Load patient-specific data set
Extract features
Apply the learned model to compute output measures of interest

Fig. 3. Overall workflow of the proposed deep learning based model.

The personalized measures of interest determined after running the personalization are:

- Systemic circulation: dead volume of the left ventricle, time at maximum left ventricular elastance, systemic resistance, systemic compliance, ratio of proximal to distal resistance of systemic circulation
- Pulmonary circulation: dead volume of the right ventricle, time at maximum right ventricular elastance, pulmonary resistance, pulmonary compliance, ratio of proximal to distal resistance of pulmonary circulation

The fully automatic optimization-based calibration method mentioned above is formulated as a numerical optimization problem, the goal of which is to find a set of parameter values for which a set of objectives are met. The number of parameters to be estimated is set equal to the number of objectives, and, thus, the parameter estimation problem becomes a problem of finding the root for a system of nonlinear equations. To solve the system of equations, we use the dogleg trust region method [18].

*3) Deep Neural Network for Real-Time Hemodynamic Analysis:* While the lumped parameter model is computationally very efficient, its personalization requires hundreds of forward runs, leading to an overall computation time of 30 – 60 seconds for determining the patient-specific measures of interest. Thus, a model based on a deep neural network capable of outputting in real time the measures of interest that would otherwise be determined using the WBC model would be a useful tool, even when run under plaintext conditions.

To train such a model, a large training database is required. Since we did not have access to a large database of patient-specific datasets we employed a strategy introduced in the past for real time diagnosis of coronary artery disease [18]: the deep neural network is trained offline on a large database of synthetically generated datasets. For each dataset, the personalization framework is run with the WBC model to determine the output measures of interest. The prediction phase is an online process, whereby the algorithm computes the measures of interest for a given patient's data, by using the learned mapping from the training phase. A schematic of the workflow is shown in Figure 3.

We comprised a database of 10000 synthetically generated input datasets which reflect the anatomical and functional variations representative of healthy and pathologic patients. Following the standard approaches, 8000 datasets are used for training and 2000 datasets for testing. The input parameters are sampled in a priori specified ranges derived from published literature [19]: the values have been selected to cover a broad range, ensuring that a wide array of anatomical variations and their corresponding hemodynamics is covered. Additional rules were defined to ensure that the datasets are physiologically sound, e.g. left and right ventricular similar stroke volume.

All input parameters were used as features for the network, leading to an input feature vector of 9 floating-point values. For more stable and faster training, features were rescaled to have the properties of a standard normal distribution with mean 0 and standard deviation of 1.

The goal is to maintain data privacy while still allowing for computations within a neural network to be successfully performed over the encrypted version of the data. Hence, both the input feature vector and the target parameters were encrypted following the MORE encryption strategy.

A fully connected neural network with 3 hidden layers was then trained to minimize the L2 distance between the 12 estimated parameters, in the encrypted form, and their corresponding targets, as well in the encrypted form. As non-linearities, both the logistic sigmoid and hyperbolic tangent (tanh) functions were chosen, with all 3 hidden layers holding the same number of neurons (40). As the problem being solved is formulated as a regression task, no activation function was set in the output layer, every output value being a linear combination of the outgoing values of the last hidden layer. The proposed network is summarized as follows: input(9) - fc(40) - tanh - fc(40) - tanh - fc(40) - sigmoid - output(12), where the numbers in the parentheses represent the number of total units in the layers.

Training was performed in batches of 32 data entries, for a number of 4500 epochs, following the SGD optimization strategy, with a chosen learning rate of 0.01.

## V. Results

### A. Performance

A first goal was to verify the correctness of the computations. Hence, in the following we present results by running the algorithms with unencrypted data (plaintext) and encrypted data (ciphertext). Note that for consistency and for enabling a fair comparison, the same hyper-parameters and random initializations were adopted.

A common question raised while training neural networks is when to stop the training to achieve the optimal performance. While an insufficient training may result in non-optimal results by underfitting the data, a too long training phase may lead to overfitting, which again can result in poor performance on the unseen dataset. A typical strategy is to closely monitoring both the training and the validation losses and to stop the training when the first signs of overfitting are observed. Alternatively, the number of epochs may be set to an arbitrary large number, and the training is stopped if the validation loss does not improve for a certain number of epochs. While both strategies are straightforward to implement during training on plaintext data, they become impractical when dealing with ciphertext data. In the latter case, the loss becomes encrypted, and if two encrypted numbers are compared, the result is also a ciphertext, which cannot be used inside a conditional statement. The inconvenience of not seeing the actual loss value forces the training to take place for a predefined number of epochs.

As the overall goal of the study is to assess the feasibility of the deep neural network to operate directly on ciphertext data, i.e. showing that the performance does not drop compared to the plaintext setting, we have chosen an arbitrarily large number of epochs to conduct the experiments and report the performance.

All experiments indicated that the training progresses similarly in both the encrypted and the unencrypted use cases, as is outlined in the following.

*1) MNIST classification:* The most important metric is the absolute accuracy of the classification models, i.e. the percentage of correctly labeled digit images. To compute the metric, the outputs of the model outputting ciphertext results are decrypted with the symmetric key. The unencrypted network achieved a classification accuracy of 98.3% on the testing dataset, which is preserved by the encrypted network.

While 98.3% is a marginally acceptable accuracy on the MNIST dataset, it is still relatively far away from 99.77%, declared as the state of the art accuracy for the digit recognition task. However, this is not surprising, as the network proposed to solve the classification task was chosen not with the intention of improving recognition accuracy, but rather to validate privacy-preserving computations in the context of neural network models. The accuracy of any predictive model generally improves with more favorable activation functions and optimization algorithms.

*2) Hemodynamic Analysis:* We validated the encrypted model at two levels: (i) at training level, in terms of its capability of preserving the correctness of the computations, and (ii) at inference level, where the focus lies on the overall capability of the model to estimate the outputs of the whole body circulation model.

To show the ability of the network to learn from ciphertext

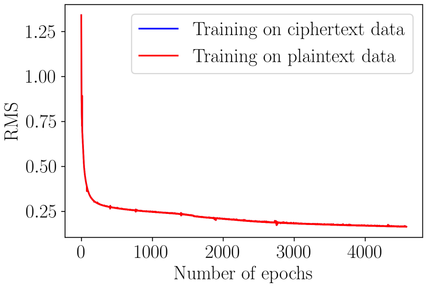

Fig. 4. Training loss evolution for encrypted and unencrypted networks. Differences between learning curves caused by floating point arithmetic are unnoticeable.

TABLE I

Results of the Deep Neural Network for Real-Time Hemodynamic Analysis on the testing dataset.

| Circulation | Parameters | MAPE [%] | Pearson correlation [%] |
|---|---|---|---|
| Systemic | Dead volume | 7.03 | 0.9997 |
| | Time at max. elastance | 0.13 | 0.9995 |
| | Resistance | 0.17 | 0.9999 |
| | Compliance | 2.45 | 0.9867 |
| Pulmonary | Dead volume | 9.88 | 0.9991 |
| | Time at max. elastance | 0.10 | 0.9994 |
| | Resistance | 0.32 | 0.9998 |
| | Compliance | 0.67 | 0.9983 |

data, the training loss, as resulted after decryption, is depicted in Figure 4.

To evaluate the hemodynamic results we use the symmetric key to encrypt the testing input feature vectors, feed the ciphertexts to the trained encrypted model and then collect the encrypted results. Similar to the MNIST digit recognition use case, we decrypt the results before evaluating the performance. Only at this point we compute the metrics for the decrypted results. To demonstrate the capability of the trained model to estimate the outputs, we computed the mean absolute relative error and the Pearson correlation, and display the results in Table I. Scatter plots of the measured versus predicted parameters for highest and lowest correlation coefficient are presented in Figure 5 and 6. The first one displays the results of the encrypted neural network model for estimating the ratio of proximal to distal resistance in the systemic circulation. The latter presents the results for systemic resistance prediction. The outputs of the model trained on ciphertext data are statistically non-distinguishable from those obtained with the model whose input feature vectors were represented as plaintext data.

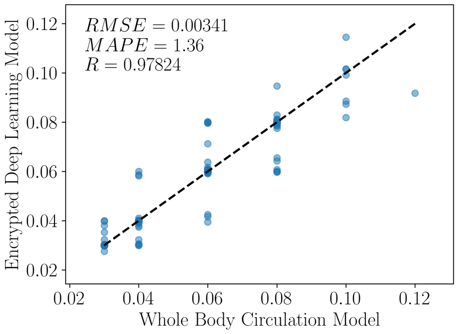

Fig. 5. Predicted versus ground truth ratio of proximal to distal resistance in the systemic circulation.

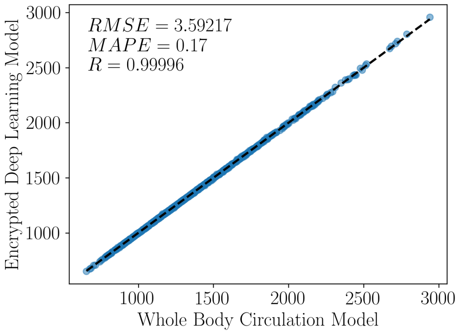

Fig. 6. Predicted versus ground truth systemic resistance.

## B. Execution time

All runtimes reported in the current section were measured on a machine equipped with an Intel(R) Xeon(R) CPU running at 2.10GHz. The deep learning library which integrates the MORE encryption scheme was written in C++. The library is still under active development, with minimal multi-threading support.

A detailed comparison of the runtime for each of the applications is given in Table II and Table III. Note that all results were reported under the assumption of employing data parallelism (8 threads) at training and inference level.

## C. Security concerns

While the MORE design is simple and clean, with homomorphic properties tailored to privacy-preserving deep neural network, the linear transformations used as the only component of the encryption algorithm limits the security. As stated in [12], [13], the scheme is vulnerable to the chosen plaintext attacks. In particular, if an attacker has access to a large enough number of pairs of encrypted and unencrypted messages, it is possible to compute the secret key by formulating and solving a numerical optimization problem, i.e. by finding the best fit of a matrix $S$ such that $(S^{-1}C_iS)_{1,1} = m_i$ for each known pair $(C_i, m_i)$. This key search attack cannot be applied on the original MORE scheme (on integers modulo $N$) because the modulo operation is nonlinear.

Although this methodology has weaker security than other homomorphic encryption schemes, it can still be used in applications where the key is never disclosed, e.g. a hospital encrypts the data and then uploads encrypted data to an external computing service. Similarly, it can be employed in a case where encryption is performed per patient, e.g. an application where one can upload personal medical data to a service that provides a personalized risk factor or other relevant health indices.

## VI. Discussion and Conclusions

In the past few years, the raised concern for protecting the privacy of sensitive medical data while still encouraging the delivery of personalized medicine solutions, increased the focus on enabling privacy-preserving computations inside Deep Neural Networks.

The proposed solution aims at ensuring the privacy by incorporating a data encryption mechanism and delivering reliable results, to be used in clinical workflows. We have showcased the applicability of incorporating the MORE encryption scheme into Deep Learning models by tackling two different problems: digit recognition and whole body

TABLE II

RUNTIME ANALYSIS OF THE ENCRYPTED AND PLAINTEXT CNNS FOR MNIST DIGIT RECOGNITION.

| Operation | Runtime [s] on cipertext data | Runtime [s] on plaintext data | Encrypted - Unencrypted ratio |
|---|---|---|---|
| Data encryption and key generation | 2.44±0.016 | - | - |
| Training (1 epoch) | 444.59±8.53 | 12.98±1.17 | 34.25 |
| Data encryption | 0.39±0.009 | - | - |
| Inference (10K images) | 20.42±0.32 | 0.54±0.08 | 37.81 |
| Data decryption | 0.001±0.0005 | - | - |

TABLE III

RUNTIME ANALYSIS OF THE ENCRYPTED AND PLAINTEXT FCN FOR WHOLE BODY CIRCULATION HEMODYNAMIC ANALYSIS.

| Operation | Runtime [s] on cipertext data | Runtime [s] on plaintext data | Encrypted - Unencrypted ratio |
|---|---|---|---|
| Training (1 epoch) | 0.66±0.09 | 0.021±0.001 | 31.4 |
| Inference (2000 samples) | 0.102±0.01 | 0.006±0.0009 | 17 |

hemodynamic analysis. We have addressed both the training and the inference phase, and showed that both can be performed on encrypted data. We demonstrated that the accuracy of the encrypted model is statistically not discernable from the unencrypted model, and that, by following the proposed strategy, computations over ciphertext data are only slightly more costly than the ones performed on plaintext data.

In conclusion, we showed that by employing the MORE fully homomorphic encryption scheme as a privacy preserving mechanism, we enabled the application of Deep Learning models on encrypted data without compromising the accuracy at all. Although the runtime increased by more than one order of magnitude, the encrypted models are still outputting results in a reasonable amount of time. With its direct support for computations over rational numbers, and the ability to perform operations without adding noise, the scheme becomes eligible for more complex models from the realm of Deep Learning.

Although the MORE encryption scheme is an attractive choice due to its unbiased advantages in terms of performance and usability, we acknowledge that it offers a lower security compared to standard schemes, and it is by no means a definitive option for problems requiring homomorphic encryption. Improving the security while maintaining the performance and potential to be used in real-world applications represents our main future work direction.

## VII. ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union's Horizon 2020 Programme (H2020/2014-2020) under grant agreement n° 732907.

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CCCDI – UEFISCDI, project number ERANET-FLAG - CONVERGENCE (2), within PNCDI III.

## REFERENCES

[1] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 05 2017. [Online]. Available: https://dx.doi.org/10.1093/bib/bbx044

[2] Z. Obermeyer and E. J. Emanuel, "Predicting the future - big data, machine learning, and clinical medicine." *The New England journal of medicine*, vol. 375 13, pp. 1216–9, 2016.

[3] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15. New York, NY, USA: ACM, 2015, pp. 1310–1321. [Online]. Available: http://doi.acm.org/10.1145/2810103.2813687

[4] A. Kipnis and E. Hibshoosh, "Efficient methods for practical fully homomorphic symmetric-key encrypton, randomization and verification," *IACR Cryptology ePrint Archive*, vol. 2012, p. 637, 2012.

[5] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. E. Lauter, M. Naehrig, and J. R. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *ICML*, 2016.

[6] E. Hesamifard, H. Takabi, and M. Ghasemi, "Cryptodl: Deep neural networks over encrypted data," *CoRR*, vol. abs/1711.05189, 2017. [Online]. Available: http://arxiv.org/abs/1711.05189

[7] H. Chabanne, A. de Wargny, J. Milgram, C. Morel, and E. Prouff, "Privacy-preserving classification on deep neural network," *IACR Cryptology ePrint Archive*, vol. 2017, p. 35, 2017.

[8] J. Mancuso. Privacy-preserving machine learning 2018: A year in review. [Online]. Available: https://medium.com/dropoutlabs/privacy-preserving-machine-learning-2018-a-year-in-review-b6345a95ae0f

[9] H. Chung and M. Kim, "Encoding rational numbers for fhe-based applications." *IACR Cryptology ePrint Archive*, vol. 2016, p. 344, 2016.

[10] C. Gentry and S. Halevi, "Implementing gentry's fully-homomorphic encryption scheme," in *CRYPTO 2011*, 2010.

[11] A. El-Yahyaoui and M. D. Elkettani, "Fully homomorphic encryption: state of art and comparison," *International Journal of Computer Science and Information Security*, vol. 14, no. 4, p. 159, 2016.

[12] D. Vizár and S. Vaudenay, "Cryptanalysis of chosen symmetric homomorphic schemes," in *CRYPTO 2014*, 2014.

[13] B. Tsaban and N. Lifshitz, "Cryptanalysis of the more symmetric key fully homomorphic encryption scheme," *J. Mathematical Cryptology*, vol. 9, pp. 75–78, 2014.

[14] L. Deng, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Processing Magazine*, vol. 29, pp. 141–142, 2012.

[15] V. Mihalef, L. Itu, T. Mansi, and P. Sharma, *Lumped Parameter Whole Body Circulation Modelling*. Cham: Springer International Publishing, 2017, pp. 111–152.

[16] S. G. Shroff, J. S. Janicki, and K. T. Weber, "Evidence and quantitation of left ventricular systolic resistance." *The American journal of physiology*, vol. 249 2 Pt 2, pp. H358–70, 1985.

[17] L. M. Itu, P. Sharma, B. Georgescu, A. Kamen, C. Suciu, and D. Comaniciu, "Model based non-invasive estimation of pv loop from echocardiography," *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6774–6777, 2014.

[18] L. M. Itu, S. Rapaka, T. Passerini, B. Georgescu, C. Schwemmer, M. Schoebinger, T. G. Flohr, P. Sharma, and D. Comaniciu, "A machine-learning approach for computation of fractional flow reserve from coronary computed tomography." *Journal of applied physiology*, vol. 121 1, pp. 42–52, 2016.

[19] D. Mann, D. Zipes, P. Libby, and R. Bonow, *Braunwald's Heart Disease E-Book: A Textbook of Cardiovascular Medicine*. Elsevier Health Sciences, 2014. [Online]. Available: https://books.google.de/books?id=1R44BAAAQBAJ