Towards Privacy-Preserving Deep Learning based Medical Imaging Applications

Anamaria Vizitiu*, Cosmin Ioan Nită*, Andrei Puiu*, Constantin Suciu†, Lucian Mihai Itu†

*Department of Automation and Information Technology
Transilvania University of Brașov, Brașov, Romania
†Corporate Technology
Siemens SRL, Brașov, Romania

Abstract—Following the reports of breakthrough performances, machine learning based applications have become very popular in the medical field. However, with the recent increase in concerns related to data privacy, and the publication of specific regulations (e.g. GDPR), the development and, thus, exploitation of deep learning based applications in clinical decision making processes, has been rendered impossible in many cases. Herein, we describe and evaluate an approach that employs Fully Homomorphic Encryption for allowing computations to be performed on sensitive data. Specifically, the solution exploits the MORE scheme and does not disclose patient data. The chosen encryption scheme increases the runtime only marginally and, importantly, allows for operations to be performed directly on floating point numbers, which represents a critical property for artificial neural networks. The feasibility and performance are first evaluated on a standard benchmarking application (MNIST digit classification). Next, we considered a medical imaging application, i.e. classification of coronary views in X-ray angiography. The reported results indicate that the proposed solution has great potential: (i) computational results are indistinguishable from those obtained with the unencrypted variants of the deep learning based applications, and (ii) run times increase only marginally. Finally, we also discuss in detail security concerns, and emphasize that the proposed solution may be employed in several practical applications, while still significant limitations remain to be solved in future work.

Index Terms—Homomorphic encryption, Deep Learning, medical data, privacy

I. INTRODUCTION

Data-driven approaches are currently being used in the medical field for various tasks related to patient diagnosis, treatment planning and disease prevention [1]. Specifically, machine learning gained a lot of traction since it provided unprecedented results on problems requiring artificial intelligence.

However, the increasingly high potential of such data-driven methods leads to a high demand for medical data which, in turn, leads to security concerns. Compared to other types of data, medical data is often associated with personal information which is subject to laws and regulations, rendering it unusable without consent from the patient. This concern has been raised and recently became a debated subject [2]. Also, the currently adopted regulations (e.g. GDPR in EU, HIPAA in USA) further demand for privacy-preserving solutions that enable both security and data usability.

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

A typical solution for enabling access to medical data while retaining security consists in stripping off all personal information through anonymization. Unfortunately, it is not always a viable solution, as patient data is often unstructured, requiring in many cases manual steps. Furthermore, it sometimes requires removing or altering properties that can provide relevant information for the machine learning model, e.g. age, location, ethnicity. An alternative solution to data anonymization consists in using homomorphic encryption (HE): a special type of encryption that allows operations to be performed directly on encrypted data without decrypting and, therefore, without having access to the real data. In this case, a machine learning approach consists of both training on encrypted data and producing encrypted results, therefore securing both the input and output data.

A typical workflow based on machine learning and HE is displayed in figure 1. In this basic scenario, a client uploads sensitive data to an external computing service which performs a training operation. Before being uploaded, data is encrypted locally with a the secret key that is never shared. Therefore, the external computing service does not see the actual data but only its encrypted version. Since homomorphic encryption is employed, the computing service is allowed to perform operations on the data, including the training of a deep learning model. This results in a model that provides encrypted predictions, which can only be decrypted by the client with the secret key. All these operations are performed by the computing service without learning anything about the data.
Employing HE in deep learning applications (or any application) is a notoriously difficult problem as HE is subjected to several limitations: (i) to our knowledge all of the existing HE schemes only allow computations over integer numbers, rendering it unusable for most real-world applications, (ii) performing operations over encrypted data is several orders of magnitude slower than performing operations over plaintext data, and, finally, (iii) operations performed on encrypted data add noise, therefore only a small sequence of operations can be performed. To address these limitations we propose a variant of the MORE encryption scheme [3] that directly operates on real numbers, and is fully homomorphic with respect to algebraic operations. Furthermore, we show that it is possible to also evaluate non-linear functions as required in a deep learning application. We evaluate the proposed methodology on two benchmark problems: the classic MNIST digit classification problem, and a personalized medicine application.

The rest of the paper is organized as follows: Section II reviews the current achievements in the domain; Section III provides an overview of the homomorphic encryption scheme; Section IV addresses the proposed privacy-preserving pipeline in two deep learning applications; Section V presents the results, with focus on the comparison with the unencrypted version; and, finally, Section VI draws the conclusions and outlines a set of questions that remain to be addressed in future work.

II. RELATED WORK

In the past few years, great effort has been invested in the development of different fully homomorphic encryption schemes [4] with the potential of bridging the gap between data security and data utility, demanded by the recent rise of privacy concerning scenarios. Most of the encryption schemes that followed Gentry’s first fully homomorphic scheme [5] are known for their efficiency in terms of security, but they suffer greatly from run-time bloat and noise, which clearly restrains their usability in real-world applications. With computations being several orders of magnitude slower than the plaintext counterparts, the accumulated noise that limits the overall number of operations that can be performed without altering too much the performance, and all operations being performed modulo \( N \), they pose a great challenge for the synergy of machine learning and data analysis. While recent advances in homomorphic encryption lead to many variants of encryption schemes, there is no currently available scheme that can manipulate rational numbers.

As a consequence, a variant of a matrix-based method, called MORE (Matrix Operation for Randomization or Encryption) [3] was adapted in the current work. The advantages introduced by MORE, over currently studied schemes, in the context of privacy-preserving networks [6], [7], [8], include: (i) low overhead, (ii) a noise-free behavior (no limitation imposed by the number of operations that can be performed on ciphertext data), and, (iii) a non-deterministic property (different ciphertexts are obtained by encrypting the same message multiple times with the same key). Moreover, the scheme supports both division and multiplication operations over encrypted data. Following the typical approach of fully homomorphic or partially homomorphic encryption schemes, the original MORE scheme applies the encryption to positive integer numbers modulo \( N \), and all the operations are performed modulo \( N \). The most important advantage of the MORE scheme, and the main claim behind the choice of this scheme, is the direct support for operations over rational numbers, that can be infused in the scheme, without the need of any encoding operation. A drawback of this extension is that the method becomes vulnerable to known ciphertext attacks, as described in Section V-C.

With recent advances in homomorphic encryption and the increased interest for data privacy, fully homomorphic encryption has been introduced as a potential solution for privacy-preserving computations within deep learning models. While in theory, the solutions are promising, in practical applications, there are still some challenges imposed by long running times, weak security and poor generalization that have to be overcome [9].

III. ENCRYPTION OF RATIONAL NUMBERS

Knowing that computations in the context of deep learning applications rely extensively on floating point arithmetic, the first step towards a privacy-preserving deep learning solution is represented by a homomorphic encryption scheme that can be directly applied to rational numbers. As a way of enabling computations to be performed on real-data, approaches that extend existing methodologies to rational numbers through an encoding mechanism have been proposed in the literature [10]. In such cases, a rational number is first encoded as an integer, or as a sequence of integer numbers, and, only then the encryption is applied on the new representation. More specifically, a rational number can be turned into an integer by multiplying it with a large scale factor, or by considering a continued fractions encoding. However performing operations on numbers represented under these forms not only directly affects the outcome of the computations, but also introduces a clear limitation in their usability. In most of the cases, the encoding mechanism also requires a noise-management technique in order to maintain the noise level below a certain threshold. Even though this seems like a straightforward problem, no reliable solution has been found to date in the context of homomorphic encryption.

Herein, to address this limitation, we have employed a variant of the MORE encryption scheme which is adapted to directly operate on floating point values. Although criticized due to weaker security [11], [12], the class of homomorphic methods based on linear transformations appears to be the only practical approach for performing privacy-preserving computations in real-world applications. Following the MORE approach, a numerical value is encrypted as a matrix and matrix algebra is employed to provide a fully homomorphic behavior, which satisfies both addition and multiplication properties. As a consequence, all operations performed on ciphertext data become matrix operations, e.g. addition of plaintext scalars.
will result in the addition of ciphertext matrices. The matrix order represents a parameter controlling the trade-off between security and efficiency: by increasing the scheme complexity (i.e. the order of the regular matrix used to encrypt a message) security is improved at a cost of slightly longer running times. For simplicity, the 2 by 2 setup is summarized in Table I.

A. Operations over encrypted data

The MORE scheme allows for algebraic operations to be performed on encrypted matrices, 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 $C_1 + C_2 = SM_1S^{-1} + SM_2S^{-1} = S(M_1 + M_2)S^{-1}$. Similarly, this property holds true for subtraction and division, but also for operations involving unencrypted scalars, making the scheme fully homomorphic with respect to algebraic operations.

In real-world applications, including deep learning based approaches, non-linear (e.g. exponential, logarithmic, square root, etc) functions need to be performed. When an encryption scheme is constrained on using only algebraic operations, the typical approach to support a broader spectrum of operations involves an approximation operation of the non-linear function by finite polynomial series (e.g. truncated Taylor series). Leveraging the matrix algebra property on which the MORE scheme relies, most of the non-linear functions used in neural networks can be directly computed as matrix functions.

Moreover, this approach is not bounded to non-linear functions, comparison operations between an encrypted matrix $C$ and a plain scalar $s$ being possible. The only limitation of the scheme refers to non-linear binary operations involving two encrypted values. We emphasize that in deep learning based applications these types of operations can be completely bypassed.

IV. DEEP NEURAL NETWORKS OVER ENCRYPTED DATA

In this section, we evaluate the correctness of privacy-preserving computations in two imaging problems: a simple problem taken from the computer vision field, and one with a clinically realistic use case. Thus, we first address a multi-class digit classification problem as a benchmarking application, and then focus on training a neural network model to solve a binary classification problem on encrypted coronary angiography views.

Experiments demonstrate that the proposed privacy-preserving neural networks can be efficiently employed as a powerful data analysis tool, by providing meaningful results, while ensuring the security of personal data.

A. MNIST: a typical dataset for neural networks

To evaluate the use of deep neural network models over ciphertext data, we consider the classification of handwritten digit images, as provided by the MNIST (Modified National Institute of Standards and Technology database) dataset [13]. The data includes 60,000 grayscale images, each having 28x28 pixels in size, alongside the appropriate label of the digit depicted in the sample.

We solve the privacy-preserving classification problem using a convolutional neural network (CNN) model where a training sample is represented as a pair of input-output ciphertext data, which corresponds to the encrypted version of the image and the associated label. The CNN considered for the digit recognition task consists of 6 layers: conv(8)-pool-conv(16)-pool-fc(100)-fc(10). Both convolutional layers have small 3 × 3 filters, with the first layer holding 8 filter maps, while 16 are being generated by the second layer. For dimension reduction, an average pooling layer is added after each convolution. Sigmoid activation functions were introduced for non-linearity into the network, except for the last layer were a Softmax function was considered to interpret the outputs as probabilities.

First, a secret key $S$ is generated, which is then used to encrypt all training samples, following the MORE encryption scheme strategy. Knowing that a scalar message $m$ is encoded as a $M \in \mathbb{R}^{2x2}$ matrix, the resulting training dataset consists of ciphertexts from the $\mathbb{R}^{28x28x2x2}$ and $\mathbb{R}^{10x2x2}$ domains. Finally, with the homomorphic property underlying MORE, and the direct support for floating-point arithmetic, training a CNN to recognize encrypted digits can be performed similarly to traditional unencrypted approaches. Hence, training was conducted on batches of 32 samples to minimize a cross entropy loss between target and predictions, with both input and output represented as ciphertext data, following a stochastic gradient.
In a 256x256 pixel resolution. We have conducted multiple coronary angiography images by a factor of 2, resulting and zooming. For runtime efficiency, we down-sampled the objectives:

- Anatomical assessment: automatically determining the anatomical severity of stenoses.
- Non-invasive functional assessment: automatically computing functional diagnostic indices. [18], [19].
- Reporting: composing medical reports automatically based on the findings in the coronary angiographies.

In view of the limitations of the pure anatomical evaluation of CAD, the functional index of Fractional Flow Reserve (FFR) has been introduced as an alternative [16], and recent technological advances also allow for image-based functional assessment of coronary stenoses based on ICA [17], [18], [19]. Coronary angiographies are recorded separately and sequentially for the right coronary artery (RCA) and the left coronary artery (LCA) (Figure 2).

An important research area in CAD is the fully automated post-processing of coronary angiographies [20], having as objectives:

- Anatomical assessment: automatically determining the anatomical severity of stenoses.
- Non-invasive functional assessment: automatically computing functional diagnostic indices. [18], [19].
- Reporting: composing medical reports automatically based on the findings in the coronary angiographies.

In this and other clinical settings based on the use of ICA, automatic LCA/RCA view classification represents an important pre-processing step. In the following section, we describe our approach for automatic coronary angiography view classification. We considered a dataset composed from 3378 coronary angiographies, which were manually annotated as displaying the LCA or the RCA. For each angiography we extracted automatically one frame, in which the arteries were well visible. The dataset was split into 1996 samples for training and 680 for validation, while 702 images were kept for the final testing of the trained model (splitting was performed at patient level, i.e. ensuring that all coronary angiographies of a patient are put in the same dataset - train, validation or test). All 3 datasets were balanced, with a 1:1 prevalence for LCA and RCA cases.

Moreover, to limit overfitting, aside from the regularization added into the network, we also performed an offline augmentation, increasing the size of the training dataset by a factor of 4. As augmentation strategies we adopted transformations involving rotating the images by ±10 degrees, shifting and zooming. For runtime efficiency, we down-sampled the coronary angiography images by a factor of 2, resulting in a 256x256 pixel resolution. We have conducted multiple experiments and concluded that for coronary angiography view classification, images having the original 512x512 resolution do not improve the classification accuracy.

Since we are dealing with sensitive data, we focused on training the CNN network on encrypted data. We chose to encrypt only the input data, i.e. the coronary angiography images, and leave the target, i.e. binary label 0 or 1, as plaintext to show that training can as well be performed if labels are kept unencrypted. Note that training can be performed by also encrypting the target, as shown in the multi-class classification problem on the MNIST dataset.

For classifying X-ray coronary angiographies we adopted the following topology of the CNN network:

- **Convolution**: 4 filters, 3x3 kernel, sigmoid function.
- **Average pooling**: stride of 2x2.
- **Convolution**: 8 filters, 3x3 kernel, tanh function.
- **Average pooling**: stride of 2x2.
- **Convolution**: 16 filters, 3x3 kernel, tanh function.
- **Average pooling**: stride of 2x2.
- **Convolution**: 32 filters, 3x3 kernel, tanh function.
- **Average pooling**: stride of 2x2.
- **Fully connected**: 64 units and tanh activation function.
- **Dropout**: dropout rate set to 25%.
- **Fully connected**: 1 unit and sigmoid activation function.

We have set the learning rate to 0.01 and then trained the network on mini-batches of 16 images, over 100 training epochs, to solve a binary classification problem which minimizes a cross entropy loss. Once the training is finalized, the encrypted form of the model can be employed to predict new encrypted instances, where angiographic images are encrypted with the same key as the ones used during the training phase.

### V. Results

For each of the tasks, we compared both the unencrypted version and the counterpart encrypted version. While the first experiment implies regular training and inference operations, for the encrypted version the neural network exclusively operates on ciphertext data (all trainable parameters are completely encrypted). To enable a fair comparison, all networks (plaintext and ciphertext) were trained using the same training strategy, hyper-parameters, and initialization.

Typically, for optimal performance, on the unseen data samples, the training is closely monitored. Since the proposed encryption scheme doesn’t support comparison between
two encrypted numbers, strategies such as early stopping or cross-validation, used during training to avoid overfitting, are impracticable. By training exclusively on plaintext data, the error metric becomes encrypted. This means that a proper stopping criterion has to be pre-defined. Hence, for usability and simplicity, we have chosen an arbitrarily large number of epochs to conduct the experiments and report the performance.

Further, the performance of the proposed algorithms is evaluated from two perspectives: validity and applicability in clinical scenarios. We first analyze the ability of the model in preserving classification accuracy, by comparing the results obtained when running the trained algorithms over unencrypted data (plaintext) against the encrypted data (ciphertext) version. Secondly, to assess the feasibility of employing privacy-preserving deep neural networks for operating exclusively on ciphertext data in a clinical routine, we report both the inference and the training running time.

All the algorithms were implemented in C++ and executed on a machine equipped with an Intel(R) Xeon(R) CPU running at 2.10GHz. The neural network operations and MORE encryption computations were exclusively CPU based. The library has support for minimal multi-threading option, and GPU support is currently under development.

A. Performance based comparison

1) MNIST classification: During inference, digit images are fed to the network in their encrypted form to generate encrypted results, which are then decrypted with the symmetric key before computing the metric. With respect to the model classification accuracy, the encrypted network achieved a 98.3% performance.

State of the art neural network approaches on the MNIST dataset have achieved near-perfect performance, being able to correctly classify 99.77% of the test cases. The difference in accuracy stems from the design choice of the network architecture, more stable optimization algorithms and optimal activation functions. Here, the digit recognition model was proposed more as a validation example for privacy-preserved computations in deep network models, and not with the purpose of improving the state-of-the-art digit recognition accuracy.

2) X-ray coronary angiographies classification: To show the ability of the network to learn from ciphertext data, the training and validation accuracy, as resulted after decryption, are depicted in Figure 3.

Regarding the classification accuracy, the CNN network trained on ciphertext data achieves 96.2% of correctly classified samples, when evaluated on unseen encrypted angiographies. When compared to the unencrypted model, accuracy was identical and hence emphasizing their capability to accurately classify the X-ray coronary angiographies.

B. Runtime comparison

Table II and Table III display the mean values and standard deviations of the runtimes of various components. All reported results were obtained by employing data parallelism (8 threads), both at training and inference level.

C. Security concerns

While the MORE scheme benefits from a clean and simple design, with promising properties tailored to privacy-preserving deep neural networks, the linear transformation underlying the encryption algorithm makes the scheme vulnerable to chosen plaintext attacks [12], [11]. With access to a large enough number of pairs of encrypted and unencrypted messages, an optimization problem can be formulated in order to recover the secret key: finding the best fit of a matrix $S$ such that $(S^{-1}C_i S)_{1,1} = m_i$ for each known pair $(C_i, m_i)$.

Although MORE provides weaker security, as compared to other homomorphic encryption schemes, it remains a viable solution for applications where encryption is performed at patient level, or when the key is never disclosed. We emphasize its applicability in scenarios where data is encrypted at the hospital level, and then uploaded to an external computing service, or when an entity uploads personal medical data to a service providing personalized health indices (e.g. risk scores).

VI. DISCUSSION AND CONCLUSIONS

In recent years there was an increasingly raised concern regarding the privacy of medical data. At the same time, data-driven approaches are more and more employed in medical practice, leading to the requirement of performing a trade-off between data availability and security. We proposed a solution...
TABLE III
RUNTIME ANALYSIS OF THE ENCRYPTED AND PLAINTEXT CNNs FOR ANGIOGRAPHIC VIEW CLASSIFICATION.

| Operation | Runtime [s] on ciphertext data | Runtime [s] on plaintext data | Encrypted - Unencrypted ratio |
|-----------|-------------------------------|-------------------------------|-----------------------------|
| Training  | 1075.47±45.54                 | 34.48±1.12                    | 31.19                       |
| (1 epoch) |                               |                               |                             |
| Inference | 26.36±1.98                    | 0.8±0.06                      | 32.95                       |
| (702 images) |                             |                               |                             |

based on homomorphic encryption that promotes both security and availability, by allowing Deep Learning algorithms to be used directly on encrypted data without performing first a decryption.

We evaluated the proposed methodology on two benchmark cases: a digit recognition problem, and a medical application consisting of coronary angiography image classification. The main goal of these experiments was to demonstrate that employing homomorphic encryption for Deep Learning problems is a viable option, with potential to be used in real-world applications. Therefore, we tracked two main quantities: (i) the difference between results obtained using the classic approach (with no encryption) and results obtained using encryption, and (ii) computation time differences between the two scenarios. Results show that the numerical differences between the two cases are negligible, while the computation time is about one order of magnitude slower for the encrypted case. Although significantly slower, it is currently outstandingly faster compared to classic HE schemes where the difference is of around 7 orders of magnitude.

In conclusion, we showed that by employing the proposed variant of the MORE fully homomorphic encryption scheme as a privacy-preserving mechanism, we enabled the application of Deep Learning models on encrypted data with minimal impact on accuracy and computation time. Due to its inherent compatibility with real numbers, along with the possibility of performing an unlimited number of operations without adding noise, we suggest that a MORE based approach for homomorphic encryption is currently the only viable option that can be employed in a practical application.

While the performance and usefulness of the MORE encryption scheme prompts it as an attractive solution for data privacy in Deep Neural Networks, it is far from being the optimal solution. Due to the security concerns it raises, the main focus for our future work lies on strengthening the security of the scheme security, while still enabling its applicability in real-world applications.

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 no. 732907.

This work was supported by a grant of Ministry of Research and Innovation, CNCS - UEFISCDI, project number PN-III-P1-1.1-PD-2016-0320, within PNCDI III.