Journal of Theoretical and Applied Information Technology
30th April 2023. Vol.101. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN INTELLIGENT MORE METHOD FOR PRIVACY - PRESERVING TRAINING TECHNIQUE IN CLOUD ENVIRONMENT

R. HARI KISHORE[1], A. CHANDRA SEKHAR[2], PRAMODA PATRO[3], PRAGATHI CHAGANTI[5]

[1]Assistant Professor, Department of Mathematics, Vasavi College of Engineering, Ibrahimbagh, Hyderabad, Telangana, India 500031
Research Scholar, Department of Mathematics, GITAM Institute of Science, GITAM University, Visakhapatnam, Andhra Pradesh, India, -530045
kishore.rh6@gmail.com

[2]Department of Mathematics, GITAM Institute of Science, GITAM University, Visakhapatnam, Andhra Pradesh, India, -530045
cakkaped@gitam.edu

[3]Department of Engineering Mathematics, College of Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad, Telengana, India, 500075
pramoda.mtech09@gmail.com

[4]Associate Professor, Department of Mathematics, GITAM, Rushikonda, Visakhapatnam, Andhra Pradesh 530003, India
pchagant@gitam.edu

ABSTRACT

For internal computations and training with huge data in an acceptable period of time, traditional machine learning modelling needs a lot of computer power. Cloud computing has made this procedure easier in recent days, but it has also introduced new security risks such as data leaks. Owing to its capacity to conduct operations over ciphertext, the previous research effort offered a unique homomorphic encryption architecture. Nevertheless, the data given by one party is not always sufficient to construct a capable system using machine learning in this technique. It permits a non-trustworthy third-party resource to handle encrypted data without revealing sensitive information. MORE (Matrix Operation for Randomization and Encryption), a privacy-preserving training technique using the encryption technique suggested in this work, allows calculations inside a modified deep neural network approach to be directly conducted on floating-point data having comparatively low operational cost. Using a popular MNIST digit identification issue to assess the viability of the suggested method while modified deep learning is used to MORE homomorphic information, efficiency does not suffer. Finally, the experimental outcomes states that the proposed method obtains increased security compared to existing algorithms.

Keywords: *Classification Problem, Homomorphic Encryption; MNIST Digit Recognition Problem; MORE (Matrix Operation For Randomization And Encryption); Modified Deep Learning Model; High Security.*

1. INTRODUCTION

Machine learning has gained the interest of researchers in the recent days since it has formed the basis of popular internet applications [1]. Data classification is a challenge in ML and AI where a classifier is used to determine the class of an unknown data sample utilizing training set of given data samples. Designing a successful classification model necessitates a huge number of correct training samples, making it impossible for individuals or small companies to do it. Outsourcing data categorization to a third party [2] is the only reasonable answer to this challenge and

outsourcing data classification relieves clients of not just the need for a huge number of accurate training data samples, as well as the need for significant computing and storage resources (like individuals or small organizations). Several companies are beginning to offer online services like medical forecasting, risk analysis, image identification, and spam identification depending on ML-driven data analytics. A customer with a data classification requirement, for instance, might submit data samples to a service provider and pay for it to apply a trained ML classification system, which is generally provided as a black-box API [3]. The behaviour of elasticity and scalability on the fly cloud with the virtual environment is called the cloud, which is widely classified into public, private, and community cloud. Generally, the IaaS, PaaS and SaaS are suggested by the cloud. The data which is stored in a data centre, it creates issues over maintaining confidentiality over customer information [4]. The cloud computing industry provides significant support to the global environment in a variety of fields, including business, medicine, and defence. Data security is one of the significant features of a cloud environment.

In further details, an online user submits an instance to the provider, who then employs a classification model to return a response indicating the requesting instance's evaluation. A user may, for instance, request an online image recognition model to ensure the contents of an image [5]. Nevertheless, this strategy is complicated by growing privacy and security concerns. Because the categorization service handles the user's sensitive data in instance x, the user needs privacy assurances. Outsourcing the classification algorithm W means that the investor of the classification model loses control over W [6]. As a result, the owner needs assurances of privacy. Much of the time, the classification model's owner, the server, and the users do not have complete trust in one another, therefore confidential material should not be shared by anyone except the owner. Recent advancements in cloud computing have replaced conventional outsourcing approaches, allowing clients to access a variety of services via the Internet inflexible (such as on-demand, pay-per-use) way [7]. As a result, a new service paradigm emerges, in which a cloud server may provide data categorization to clients. The server, particularly, may remotely analyse and identify the customers' data samples using privately held training data samples [8]. Nevertheless, sending client data samples to the cloud poses privacy issues, as data processing in the cloud is frequently outsourced to untrustworthy third-party servers. Moreover, even if the server provides classification services to the client, it might not want to reveal any characteristics or aspects of its training database. Conventional encryption techniques can guarantee data confidentiality, but they have a hard time preserving data used in calculations [9]. Particularly the categorization, which is difficult to apply to encrypted data or models. As a result, while employing the advantages of new cloud computing technologies, protects the confidentiality of client data samples and server training data. The significance of cloud computing in data categorization may be divided into three categories:

i. Cloud is in charge of maintaining and updating the categorization training data set.
ii. Cloud delivers data grouping as a service for any customer over the Internet while maintaining client data confidentiality.
iii. Cloud assists customers in offloading a significant amount of processing.

Data storage and processing security are the privacy maintenance aspects of a cloud environment. Both cloud consumers and cloud servers consist of identical cloud security in which trust is a vital requirement [10]. From diverse backdrops, several local techniques are embraced using cloud computing. Numerous encryption algorithms carry data encryption in recent days. Privacy-preserving develops numerous approaches in the cloud. The grouping attribute proposes the privacy-aware access control approach for enhanced data privacy in the cloud sector [11]. The efficiency of this verification scheme is ineffective. MORE is a privacy-preserving training approach that relies on encryption and allows calculations inside a modified deep NN system to be directly conducted on floating-point data with comparatively low operational cost.

In this research work, section 2 defines some reviews of recent techniques in privacy preservation and classification problems. Section 3 presents the proposed methodology. Section 4 discusses the results and discussion. Section 5 concludes the research study.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. LITERATURE REVIEW

Here, discussed some of the recent advanced approaches in privacy preservation training and classification in machine learning problems. By utilizing a homomorphic cryptosystem, Shen et al. [12] designed secure building blocks, like secured polynomial multiplication and secured comparison, and constructed a secure SVM classification technique, that necessitates two interactions in a single iteration and doesn't require a reliable third-party. The proposed method protects the secrecy of sensitive data for every data source and SVM method factors for data analysts, according to a thorough security study. Thorough testing has shown that the proposed method is effective. Shortell et al. [13] proposed fully homomorphic encryption as a way to do signal processing using remote execution techniques while retaining data privacy. (1) expanding FHE to real numbers, (2) limiting the error linked to FHE procedure in contrast to unencrypted variation of the procedure, (3) enhancing the usability of FHE as a tool by employing GPU are three additional contributions of this paper. The contributions are demonstrated by using these concepts to two famous problems: Signal pr(brightness/contrast filter) and natural logarithm calculation. Chen et al. [14] created fully homomorphic encryption (FHE), that allows cloud servers to execute complex computations on behalf of customers without revealing their personal information. However, FHE is often regarded as a computationally expensive algorithm. Implement integer arithmetic over FHE for the first time, which is the foundation of many data aggregation functions, and evaluate the effectiveness from a practical perspective. Wang et al. [15] developed a secure multiparty computation (MPC) based privacy-preserving system for collaborative data mining and signal processing, wherein data-mining and signal processing are done in the CS domain. MPC protocols are solely employed for CS transformation and reconstruction in this system, with data mining and signal processing activities decoupled from MPC processes. When contrasted with previous efforts, the system has a lot of flexibility and scalability since the decoupling permits CS converted data to be reused and different data processing methods may be employed in these CS domains. Simultaneously, the architecture allows for privacy-preserving data storage on the cloud. Design an orthogonal matching pursuit method based on MPC and its related MPC protocol for CS reconstruction. The findings of the analysis and experiments indicate that the proposed framework is capable of providing effective privacy-conserving data mining or signal processing.

The first identified client-server data categorization protocol employing SVMs was proposed by Y. et al. [16]. For two or more class issues, the proposed technique uses PP classification. Pailler homomorphic encryption and secure two-party computing are used in the protocol. An efficient and innovative approach for securely extracting the sign of pailler encrypted numbers is at the centre of the protocol. Yu et al. [17] created three control groups to mimic 3 distinct situations depending on whether the client delivers encrypted data to a server and whether the server employs the HOPE technique. The ultimate findings states that when the server employs actual LR to create a model on encrypted data, the trained system performs similarly to a random guess, ensuring that data is kept private. Furthermore, as opposed to the earlier Logistical regression approach, the HOPE technique requires a little amount of computing time but has no noticeable performance loss. Kaaniche et al. [18] provided cryptographic defensive techniques, as well as research directions and technological developments for protecting outsourced data in cloud infrastructures. Xu et al [19] developed a unique method for achieving privacy-preserving ML with dispersed training data and huge shareable data portions. To accomplish privacy preservation, use data locality characteristic of Apache Hadoop framework and a small number of cryptographic procedures in Reduce() methods. Indicate that the proposed technique is safe in a semi-honest model and establish its scalability and accuracy using extensive experiments. Fung et al [20] developed a realistic and effective technique for finding a generalized version of data that hides sensitive data while remaining usable for classification modelling. The process of data generalization is carried out by specializing or specifying the degree of information in a top-down manner till a minimal privacy requirement is exceeded. For categorical and continuous characteristics, this top-down specialization is simple and efficient. This method makes use of the fact that data often contains duplicate categorization structures. While generalization may destroy certain structures, it also allows for the emergence of other structures that can assist. The findings indicate that classification quality may be maintained even under the strongest privacy constraints. This research has a wide range of applications in both the public and private

sectors, where information is shared for mutual advantage and productivity. Private protocols supporting Kernel Adatron and Kernel Perceptron classifiers, as well as private classification protocols are proposed by Laur et al [21].

Novel protocols encrypt their outcomes - either kernel value, classifier, or classification model - such that they can be decrypted by protocol participants deciding to do so. Also demonstrated is how to employ encrypted classifiers to estimate various characteristics of the input and the classification algorithm in a secret manner. As per conventional cryptographic definitions, novel SVM classification algorithms are the first to be verified private. In [22] authors have developed a new homomorphic encryption architecture for non-abelian rings, as well as defining homomorphism procedures in ciphertext space. Depending on the Conjugacy Search issue, the technique can provide one-way security. Following this, homomorphic encryption over a matrix ring was presented. It enables real-number encryption relying on homomorphism of a 2-order displacement matrix coding function and obtains rapid ciphertext homomorphic comparison without decrypting any in-between output of ciphertext processes. They also employ the technique to achieve privacy conservation for ML training and classification in the context of ciphertext. Proposed methods are effective for encryption/decryption and homomorphic processes, according to the study.

From the above review, it is observed that general homomorphic encryption has some disadvantages in it. Guaranteeing fairness (that everyone who is supposed to get an output gets it) in the event of numerous, participating parties is frequently challenging and involves additional machinery (like threshold decryption) and assumptions (threshold of honest parties, so on). Another issue in the multiple-participant approach is how the participants enter their values for calculation in secret. If a person has a private key, they decode the inputs and intrude on the privacy of the other. Thus, the work focuses on improving the encryption process in privacy preservation training and classification problems.

3. PROPOSED METHODOLOGY

Two major problems must be overcome in order to build an effective system for privacy-preserving outsourced classifier service. The first step is to figure out a viable mechanism for an untrusted server to classify protected cases using a protected classifier. The next goal is to decrease both classifier owners' as well as the service user's communication overhead. Here, offered MORE, a privacy-preserving training technique based on an encryption strategy that allows calculations inside a NN approach conducted on floating-point data with low operational overhead. Assume a popular MNIST digit identification issue in order to assess the feasibility of the suggested technique and demonstrate that efficiency does not suffer when DL is used to MORE homomorphic data. The proposed methodology procedure is depicted in the following figure 1.

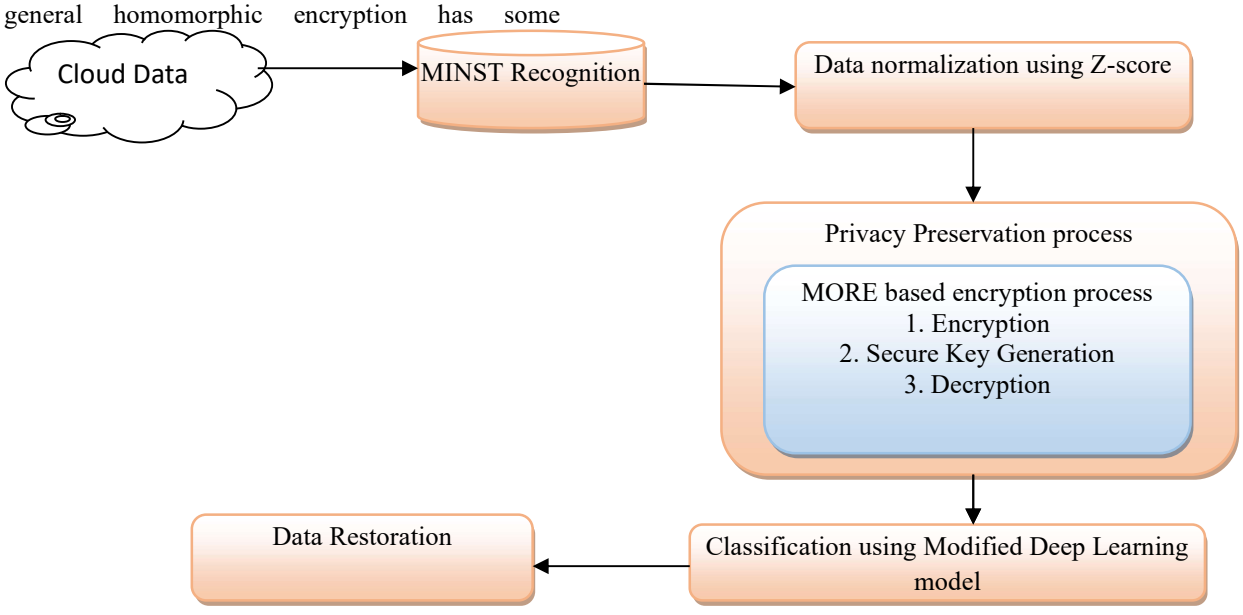

Figure 1. Proposed methodology

Journal of Theoretical and Applied Information Technology
30th April 2023. Vol.101. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.1. Problem Formulation: MNIST

Classification is a common topic explored in the NNs context. The challenge of image classification is highly particularly related to data displayed in images. MNIST [23] database comprises handwritten digits images and is commonly used as a baseline for image classifier systems. digit recognition problem was chosen for the first experiment to address the issues of privacy-preserving calculations in NN systems with the goal of giving important insights into the proposed technique's advantages and weaknesses in a real-world setting. Deep CNN models, on the other hand, have been proven to outperform other kinds of classification models on MNIST, resulting in the lowest reported test error. Furthermore, the error rate increased whenever matching shallow networks was used, underlining the necessity for models. Character identification issue is presented as estimating the likelihood that an image belongs to one of ten classes (0–9 digits). As a result, target labels are commonly expressed as one-hot vectors, with values 1 for related class and 0 for rest. This is an instance of a multiclass issue (C = 10) that may be addressed using a NN system trained to reduce cross-entropy error among predicted ($\tilde{y}_i$) and expected (y) probability distributions which is defined in the following equation 1.

$$CE(y, \tilde{y}) = -\sum_{c=1}^{c=10} y_i \log(\tilde{y}_i) \quad (1)$$

(1) DATASET

MNIST collection contains 70,000 images with a relative small size of 28×28 pixels, every image tagged with digit which shows in Figure 2. In the images, the digits are size-normalized and centred. MNIST samples were divided into 3 databases, yielding 50,000 instances for training NN classification model, 10,000 for verifying trained model, 10,000 for evaluating performance of the classification method. To prevent class imbalance difficulties that frequently occur in classification, the training data were evenly distributed among the 10 classes.

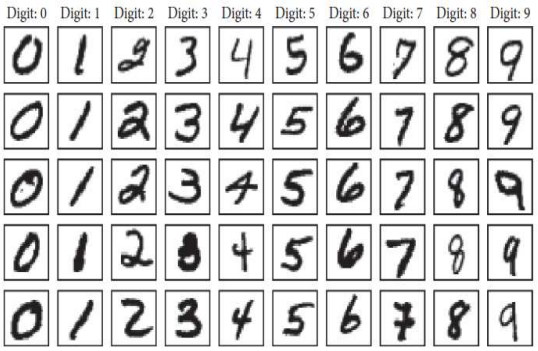

Figure 2: MNIST database sample images [23]

Pixel values in MNIST images vary from 0 to 255. Pixel values are scaled between [0, 1] depending on the least and highest pixel intensity to facilitate training convergence. MNIST labels, which were expressed by numerical values ranging [0-9], are encoded to categorical data as one-hot vectors for neural network training. As a result, every digit was indicated by the vector with a length equivalent to total classes and a value of 1 for digit position in vector, with all other values set to 0.

3.2. Data Normalization using Z-score method

The data quality given to the ML system is acutely to its accomplishment. Outliers, incorrect data types, misplaced values, immaterial features, incorrect data are common in dirty data. Any of these will prevent the ML technique from learning effectively. Correspondingly, transforming input data to usable format is an important part of the ML process. Data normalization is an ML method that necessitates converting numeric columns to a common scale. Certain feature value varies from other multiple times in ML. the learning procedure will be controlled by features with higher values. Nevertheless, these factors are more significant in predicting the model's conclusion. Data normalization converts multi-scaled to a single scale data. After normalization, entire variables have a comparable whack on the system, increasing the learning algorithm's stability and performance. Here, the Z-score approach is utilized for normalizing given data.

- **The z-score method**

The data is transformed into a distribution with a mean of 0 and SD of 1 using z-score technique (also known as standardization) [25]. By removing relevant feature's mean and then dividing by

SD(see equation 2), every standardized value is calculated.

$$x_{std} = \frac{x-\mu}{\sigma} \quad (2)$$

The feature is not rescaled to a defined range by z-score. If the data is regularly distributed, z-score generally ranges [-3.00 - 3.00] (> 99 percent of data). It's crucial to remember that z-scores aren't always evenly distributed. They simply scale the data and distribute it according to the same distribution as original data. Only when input feature follows a normal distribution will this converted distribution have a mean zero and SD one and will be standard normal distribution.

3.3. Matrix-Based Data Randomization

More than one version of MORE encryption technique is explored and developed to work directly on floating-point data. Plaintext scalar is encrypted as n × n ciphertext matrix using the MORE encryption technique [27], matrix algebra is used to permit calculations on ciphertext. As a result, every operation on ciphertext are specified as matrix operations; for example, multiplication of plaintext scalars is described as matrix multiplication of ciphertext. Sequence of matrices employed to code a message is an essential aspect in determining the security-to-efficiency trade-off. The MORE cryptosystem is presented in Table 1 for a 2 × 2 configuration.

MORE method allows to do algebraic operations on ciphertext matrices, that is., provided two encrypted matrices $C1 = SM1S- 1$ and $C2 = SM2S- 1$, for addition.

$$C_1 + C_2 = SM_1S^{-1} + SM_2S^{-1} = S(M_1 + M_2)S^{-1} \quad (3)$$

which is the encryption of $M_1 + M_2$, and for multiplication

$$C_1C_2 = SM_1S^{-1}SM_2S^{-1} = SM_1M_2S^{-1} \quad (4)$$

Subtraction and division, and also plaintext scalar operations, have the same characteristic, making the system completely homomorphic for algebraic process.

Table 1. MORE encryption method for rational numbers.

Message	Scalar value m∈ ℝ
Secret key generation	Invertible matrix $\boldsymbol{S} \in \mathbb{R}^{2*2}$
Matrix construction	$M = \begin{pmatrix} m & 0 \\ 0 & r \end{pmatrix}$, where r∈ ℝ is a random parameter
Encryption operation	Encryption$(m) = C = SMS^{-1}$
Decryption operation	Decryption$(C) = K = (S^{-1}CS)$
Message recovery	$m = K_{(1,1)}$

- **Encryption of Rational Numbers**

Actual MORE method, the same as any FHE or PHE method, is limited to positive values modulo N, with entire operations carried out modulo N. These systems rely significantly on an encoding technique to be able to work with rational numbers. Consequently, a real number is transformed to an integer (or group of integers), the technique is then employed to homomorphically encrypt encoded numbers. The use of continued fractions is a common way to formulate the encoding. Even fundamental processes on numbers stated in this form are hard to execute, despite the fact that an accurate representation may be achieved. However, by multiplying rational integers with high scaling parameters, a simpler encoding may be inserted. Though much more permissive, it necessitates a scaling factor management method, that is problematic to implement some processes, such as division, when this factor is decreased. Furthermore, by expanding the methods to work on rational numbers, noise is often introduced into the cryptosystem. As a result, a noise-control technique must be implemented to keep the noise level to a minimum. Despite the fact that dealing with rational numbers appears to be a simple task, there is presently no solution that permits them to be used in HE. MORE encryption method has the advantage of being able to be directly formulated for rational numbers.

- **Performing Operations over Encrypted Data**

With regard to basic algebraic operations, the MORE encryption method has been proven to be completely homomorphic. Nonlinear (ie, exponential, logarithmic, square root, and so on.) functions must be handled in real-world applications, with DL-based techniques. The

majority of regular nonlinear process techniques are relied on notion of approximating the stated function using a finite polynomial series (example, truncated Taylor series). Nonlinear function calculation is based purely on algebraic operations in this technique, which is perfectly compatible with the MORE encryption setup. Nevertheless, a more practical method is feasible within the MORE cryptosystem. Provided the

Algorithm 1. Sigmoid function under MORE encryption method implementation

Input: Ciphertext C ∈ $R^{2\times2}$
Output: Ciphertext R ∈ $R^{2\times2}$
(1) function Sigmoid(C) \\ Utilizing direct matrix process
(2) R←$I_2 \times (I_2 + \text{MatrixExp}(-C))^{-1}$ \\$I_2$ indicates identity matrix
(3) return R
(4) end function
(5) function Sigmoid(C) \\ Utilizing eigen decomposition
(6) L,V←Eigen Decomposition(− C)
(7) L_f←Diag(Exp(L))
(8) C_{exp}←V× L_f× V^{-1}
(9) R←$I_2 \times (I_2 + C_{exp})$− 1
(10) return R
(11) End

characteristic that govern encryption system, and knowing that ciphertext-based procedures depend on matrix algebra, nonlinear functions is calculated either (1) directly as matrix functions or (2) through matrix decomposition which a message *m,* which is encrypted, is always found amongst eigenvalues of ciphertext matrix *C.* Consider, in a 2 × 2 configuration, one of eigenvalues related to arbitrary value *r* used while matrix construction, whereas other relates to message *m*. To guarantee that message can only be recognized through appropriate decryption and by holding secret key, arbitrary value *r* is normally selected to be statistically indiscernible from message. Employing a function *f* on ciphertext data *C* is consequently equal to using *f* position. Whilst initial approach is straightforward, next method is formed on the property as said by directly on *C* eigenvalues. Hence, matrix decomposition *VLV*− 1 is initially decomposes the matrix *C* to eigenvalues *L* and eigenvectors *V*. Subsequently, nonlinear function which is to be assessed is useful independently on every of eigenvalues. At last, obtained ciphertext matrix is reformed as *Cf*=*Vf*(*L*)*V*− 1 via eigen vectors and the values assessed on function *f*. As contrasted with direct matrix function depending on calculations, this method is used to do comparisons among ciphertext matrix *C* and plain scalar *s*. either of these 2 techniques support nonlinear binary operations among two ciphertext information. Nevertheless, in DL, such process is completely dodged. Algorithm 1 depicts the sigmoid function implementation under MORE encryption method.

Starting with these techniques, Algorithm 1 demonstrates how the two proposed approaches may be used to construct the function $f(x) = (1/(1 + e^{-x}))$ specified on $x \in R$, under MORE conditions, provided any ciphertext $C \in R^{2\times2}$. The logistic sigmoid function is a nonlinear function that is usually employed in neural networks, as will be discussed in the following sections.

- **Modified Deep Neural Networks over Encrypted Data**

Features of privacy-preserving MDNNs are discussed in this section. The suggested technique utilizes MORE homomorphic encryption system, which allows conventional NN models to be trained and exploited directly on homomorphically encrypted data.

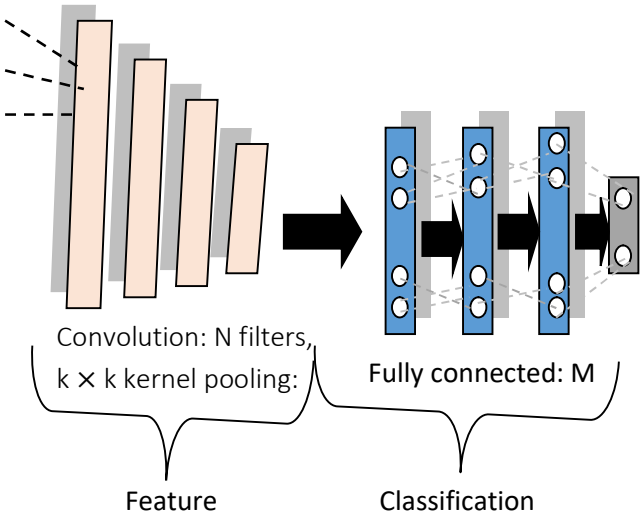

Figure 3: General structure of DCNN [23]

The above figure 3 represents conceptual representation of DCNN. Deep Learning models have a complicated mathematical formulation that ultimately comes to a sequence of repeated blocks of calculations that is dependent on a small group of basic procedures over rational numbers. Most of those modern DL-based findings [29] were produced by DNN models that used only a few types of process. Functionality of NN models is defined to account for activities on ciphertext by exploiting homomorphic characteristic of the MORE scheme. Figure 3 shows the proposed

process, which is based on deep learning. Training data is encrypted with secret key which is not disclosed before it is processed. Following that, DL-based system will only have access to encrypted data (ciphertext), whereas raw data (plaintext) will be separated from processing unit and kept private on data provider's side.

Algorithm 2: MORE secret key generation.

Iutput: Ciphertext R ∈ $R^{2\times2}$
Output: secret key **S** ∈ R2×2
(1) **function** Keygeneration( )
(2) **while** True **do**
(3) **S**⟵RandomUniform(size = (2, 2), min val, max val)
(4) **if** det(**S**) ≠ 0 **then** \\ Confirm matrix invertibility
(5) **break**
(6) **end if**
(7) **end while**
(8) **return S**
(9) **end function**

At last, the network is trained straight away on ciphertext using classical pipeline, due to homomorphic property using MORE encryption method, direct support for floating-point arithmetic, and entire process conducted inside network formulated to make sure applicability on ciphertext data. As a consequence, model generates encrypted predictions that are decoded by secret key owner. After training phase is completed, model's encrypted version is used to recognize new encrypted cases (inference stage), with input samples encrypted with similar key as training phase. Symmetric keys are used in the MORE cryptosystem. As a result, a secret key created using the technique provided in Algorithm 2 is used for plaintext data encryption and ciphertext data decryption.

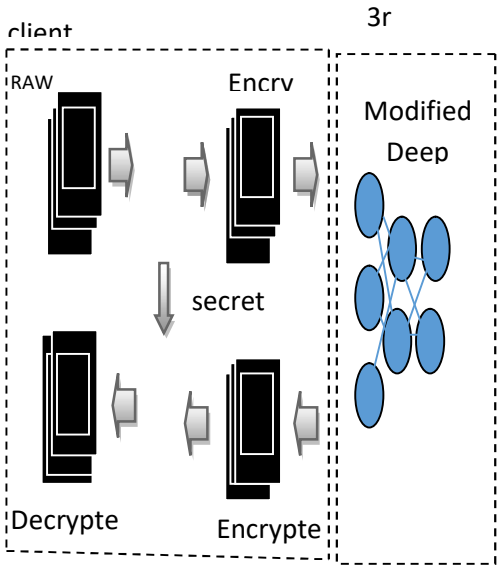

Figure 4: Process of proposed privacy-preserving MDL-based application based on homomorphic encryption.

Algorithm 3 presents the proposed MDL-based ciphertext data analysis methodology and pipeline applied on plaintext for comparison and validation. All actions done throughout training and prediction are expressed in Algorithm 3. Because outsider acts mainly on ciphertext and presents conclusions as ciphertext, data privacy is preserved throughout training and inference. As a result, medical information is securely processed such that an outside party cannot extract knowledge from patient information, as well as user cannot acquire information about the ML model.

Algorithm 3: DL-based analysis on ciphertext.

Step 1: function Train On Ciphertext( )
Step 2: $\mathbf{X}_{train}$, $\mathbf{Y}_{train}$ ← Load Dataset
Step 3: $\mathbf{X}_{train}$← Normalize($\mathbf{X}_{train}$)
Step 4: S← Key Generation
Step 5: $\mathbf{X}_{train\ enc}$ ← Encryption($\mathbf{X}_{train}$, **S**)
Step 6: $\mathbf{Y}_{train\ enc}$ ← Encryption($\mathbf{Y}_{train}$, **S**)
Step 8: Build Model()
Step 9: Train($\mathbf{X}_{train\ enc}$, $\mathbf{Y}_{train\ enc}$)
Step 10: return model $_{enc}$
Step 11: end
Step 12: function Predict On Ciphertext( )
Step 13: $\mathbf{X}_{test}$← Load Samples
Step 14: $\mathbf{X}_{test}$ ←Normalize ($\mathbf{X}_{test}$)
Step 15: S ← Load Key
Step 16: $\mathbf{X}_{test\ enc}$ ← Encryption($\mathbf{X}_{test}$, **S**)
Step 17: Load Model
Step 18: $Y_{test\ enc}$ ←Predict($X_{test\ enc}$)
Step 19: Y_{test} ← Decryption($Y_{test\ enc}$, **S**)
Step 20: return *Y* test
Step 21: end

On the other hand, DL architecture handles raw data either imagery or one dimensional signals. Further, weights are also learned in unsupervised way. After training, the activation for the individual neurons are tested using positive and negative stimulus. To further fine tune the neurons, validation can also be done using back propagation. Problems like over fitting, parameter selection can occur with DNN. So, a fuzzy inference system can minimize the aforementioned problems. Here, the neurons acts as classifiers to fed into the fuzzy-inference system.

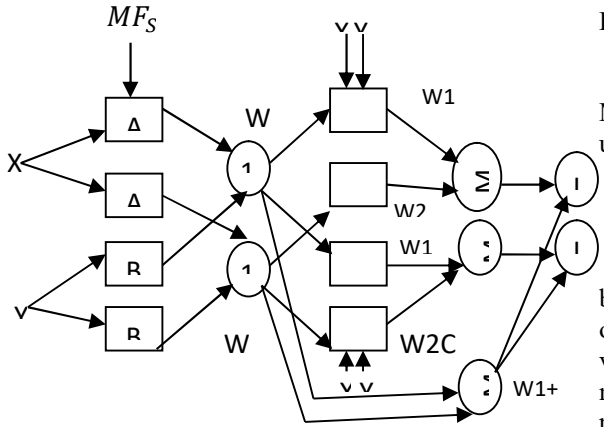

Figure.5. Process of Modified Deep Neural Network

- **Integration with Fuzzy Inference system**

Fuzzy Inference systems, allow for complex non-linear problems to be approximated using if-then statements. The advantage of structured rule-based systems is that they can be affected by subjective data. An analyst has the opportunity to improve categorization outcomes or modify the system's behaviour by offering the system specialised information [30]. Assume that there are three fuzzy if-then rules developed by Takagi and Sugeno in the rules.

Rule 1: If x is A_1, y is B_1,
and z is C_1 then $f_1=p_1x+q_1y+t_1z+ r_1$,
Rule 2:
If x is A_2, y is B_2 and z is C_2 then $f_2=p_2x+q_2y+t_2z+ r_2$,
Rule 3:
If x is A_3, y is B_3 and z is C_3 then $f_3=p_3x+q_3y+t_3z+ r_3$,

Fig. 2 illustrates the reasoning system for Sugeno model and the function of each layer is described as follows:

Layer 1 Including an adaptive node with a node function

$$O_{1,i} = \mu_{Ai}(x),\ for\ i = 1,2 \quad (5)$$
$$O_{1,i} = \mu_{Bi-2}(y),\ for\ i = 3,4 \quad (6)$$
$$O_{1,i} = \mu_{ci-4}(z),\ for\ i = 5,6 \quad (7)$$

where $\mu_{Ai}(x)$, $\mu_{Bi}(x)$and $\mu_{ci}(x)$ are any appropriate parameterized MFs and $O_{l,i}$ is the membership grade of a fuzzy set $A = (A_1, A_2, B_1, B_2\ or\ C_1, C_2)$ and It represents how well the given input, x (or y or z), complied with the quantifier which is called as ‘‘Premise Parameters’’.

Additionally, any acceptable parameterized MF, such as the generalised "bell function," can be used as the membership function for A :

$$\mu_A(x) = 1/(1 + |(x - c_i)/a_i|^{2b} \quad (8)$$

where $\{a_i, b, c_i\}$ is the parameter(s) set. The bell function changes in accordance with the values of these parameters and, as a result, displays the various MF for fuzzy set formations. This kind of membership function is usually considered in the present study.

Layer 2 Each node in this layer is fixed, that is the sum of all incoming signals determines what is produced at each node's output.

$$O_{2,i} = \mu_{Ai}(x)\mu_{Bi}(y)\mu_{ci}(z), i = 1,2,3 \quad (9)$$

Every output corresponds to a rule's "Firing Strength."

Layer 3 Includes the fixed node labelled N function of normalization:

$$O_{3,i} = \frac{w_i}{w_1+w_2+w_3},, i = 1,2,3 \quad (10)$$

Simply, outputs of this layer are called ‘‘Normalized Firing Strengths’’.

Layer 4: Adaptive nodes:

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i(p_i x + q_i y + t_i z + r_i) \quad (11)$$

It is termed "Consequent Parameters" because each node in this layer multiplies the Normalized Firing Strength output from the third layer.

Layer 5 includes a signal fixed node with the label S and a summation function that computes the DNN network's overall output as the sum of all incoming signals.

$$Overall\ output\ O_{5,i} = \sum_i \overline{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (12)$$

- **Improved Cuckoo search algorithm (ICSA)**

In this section, a basic cuckoo search (CS) method is proposed as an upgraded cuckoo search algorithm to increase the optimization capability. Symmetric design and opponent learning are combined into the CS algorithm to increase the exploitation search potential. In order to initialise the population and generate new candidate solutions in evolutionary generations, opponent learning (OL) is included into (CS). This can spread the population as widely as possible across the search space and direct it toward the more promising locations.

- **Basic CS Algorithm.**

The fundamental CS method is based on some cuckoo species' tendency to brood parasitize other host bird species by laying their eggs in their nests. The three ideal rules listed below are employed to simplify the description of the fundamental CS: (1) Each cuckoo lays one egg at a time, dropping it into a set that is selected at random; (2) The best nests with top-notch eggs will be passed down to the following generations; (3) When there are constant amounts of host nests available, the host bird has a probability of finding a cuckoo egg is $p_a \in [0, 1]$. In this scenario, the host bird has two options: either get rid of the egg or depart the nest and construct a sophisticated new nest.

This possibility reflects the outcome of each generation's replacement of cuckoo eggs (eggs found by the host bird) with new eggs. Here an egg represents a solution. These assertions guarantee that, in the selection process for the optimization method, the finest solutions will endure from generation to generation. The CS algorithm aims to swap out existing solutions in the nests for new, higher-quality ones. A new solution X_i^{t+1} for cuckoo i is given by:

$$X_i^{t+1} = X_i^t + \alpha \otimes L'evy(\lambda) \quad (13)$$
$$\alpha = \alpha_0 \otimes (X_j^t - X_i^t) \quad (14)$$

where α is the step size (α > 0) with dimension equal to the dimension of the problem; the product ⊗ represents entry-wise multiplications; X_j^t is a random selected solution; and the *Lévy*(λ) is Lévy flights random walks. The parameter α0 is chosen equal to 0.01, as recommend in, to enhance the search capability. One of the random walks that derives its step length from the Lévy distribution is called a Levy flight. The probability density function that produces the series of instantaneous jumps that characterise this distribution has a power law tail and is given by:

$$L'evy(\lambda) \approx S = t^{-\lambda}, (1 < \lambda \leq 3) \quad (15)$$

Lévy flights' step length S is selected from a uniform distribution that follows the Lévy distribution.. Additionally, the method combined a local random walk and a global exploratory random walk in a balanced manner, with a switching parameter p_a in a controlled way. The local random walk can be written as

$$X_i^{t+1} = X_i^t + \alpha s \otimes H(p_a - \varepsilon) \otimes (X_j^t - X_k^t) \quad (16)$$

where X_j^t and X_k^t are two different solutions selected randomly by random permutation, H is a Heaviside function, ε is a random number drawn from a uniform distribution, and s is the step size.

On the other hand, the global random walk is carried out by using L´evy flights:

$$X_i^{t+1} = X_i^t + \alpha \oplus L'evy(s, \lambda) \quad (17)$$

Here, $\alpha > 0$ is the step size scaling factor; L´evy(s, λ) is the step-lengths that are distributed according to the following probability distribution shown in (5) which has an infinite variance with an infinite mean:

$$L'evy(s, \lambda) = \frac{\lambda \Gamma(\lambda) \sin(\frac{\pi\lambda}{2})}{\pi} \frac{1}{s^{1+\lambda}} \quad (18)$$

- **Improved symmetric design and opponent learning based CSA**

The symmetric design and opponent learning operation are incorporated into the CS algorithm to further enhance the algorithm's search capability [26]. The fundamental goal of the symmetric design is to effectively choose the best set of levels by utilising the characteristics of the fractional experiment. An symmetric array of K factors with

Q levels and M combinations is denoted as $L_M\,(Q^K)$, where Q is the prime number, $M = Q^J$, and J is a positive integer satisfying $K = (Q^J - 1)/(Q - 1)$. The brief procedure of constructing the symmetric array $L_M\,(Q^K) = [a_{i,j}]_{M.K}$ is described in below procedure. The symmetric design process is explained in algorithm 4.

Algorithm 4: Symmetric design and opponent learning

Begin
- Build the symmetric array by using the methods listed above.
- Pick two solutions at random from the available set.
- The domain created by the two solutions should be quantized.
- Randomly generate $k - 1$ integers $p_1 \ldots p_{k-1}$
- Use $L_M\,(Q^K)$ to generate M potential offspring
- Draw a solution at random from the available options.
- The result is improved by comparison with the best solution from the symmetric M offspring.

End

Opponent Learning (OL) is a new concept in computational intelligence. The fundamental principle of OL is to take into account both a solution and its corresponding opposite solution in order to provide a more accurate approximation of the already available candidate solutions. It has been proven to be an effective method to enhance various optimization approaches. As a result, the OL concept is included in the suggested algorithm to further boost diversity and faster convergence.

Suppose $X = (x_1, x_2, \ldots, x_n)$ is a solution in an n-dimensional space, where $x_i \epsilon [Lx_i, Ux_i], (i = 1,2, \ldots, n)$. . The opposite solution $X' = (x'_1, x_{,2}, \ldots, x'_n)$ is given by:

$$x'_i = Lx_i + Ux_i - x_i \quad (19)$$

Let f (.) be a fitness function via which the fitness value can be evaluated. According to the above given definitions of X and X', if $f\,(X') \leq f\,(X)$, then X is replaced with X', otherwise X is kept. In order to determine which is best, the solution and its opposing solution are assessed simultaneously. OBL is used to start up the population and generate fresh ideas during the evolution process. The figure 6. illustrate the process of the improved cuckoo search algorithm.

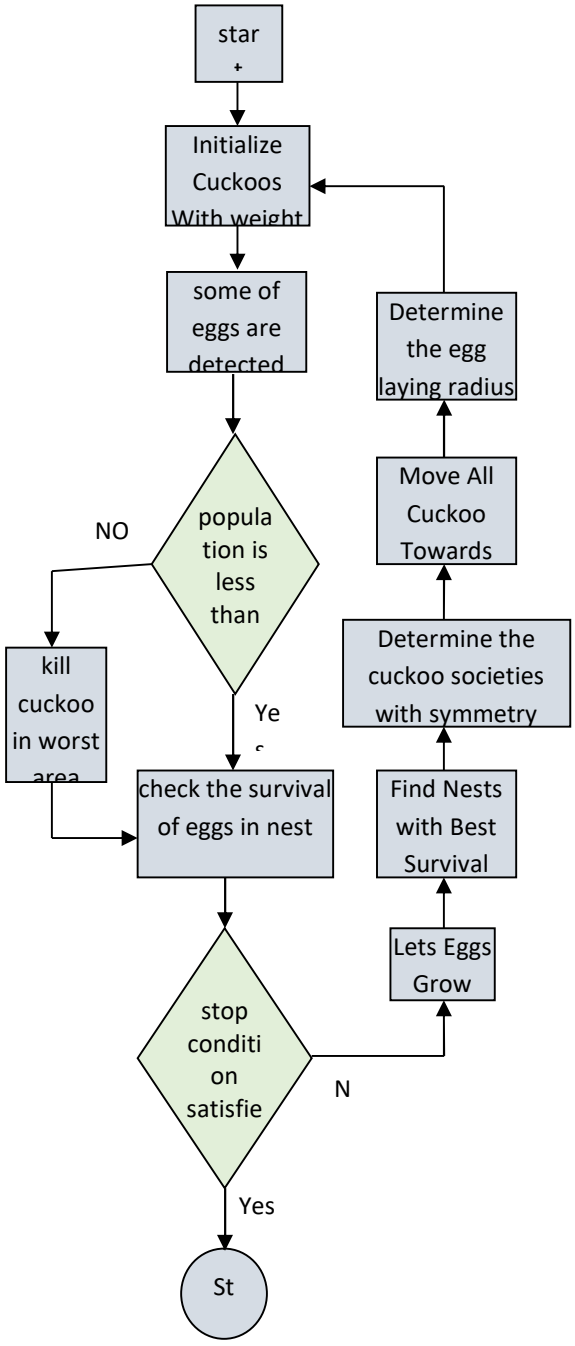

Figure 6. The process of improved cuckoo search

Thus the proposed MORE encryption method has certain benefits with respect to simplicity, clearness, practicability, with qualities targeted to privacy-preserving ML with the help of modified

deep learning model, high security when contrasted with other HE systems.

4. RESULTS AND DISCUSSION

Here, compared accuracy of proposed deep learning based MORE encryption methods with the conventional encryption methods. The absolute accuracy of different classifiers, that is., proportion of properly classified digit images, is default measure used to evaluate the classifier performance on the MNIST database. The MNIST dataset is well balanced across 10 labels, therefore accuracy may be considered a valid statistic for measuring classification performance. Even though the proposed model achieved an acceptable accuracy of 94.34 % for MNIST digit recognition test. This is expected, given that classification algorithm proposed to address digit recognition issue was originally designed to evaluate accuracy of privacy-preserving calculations inside a conventional HE.

Table 2. Comparison table for security strength between the proposed and existing methods

File size (bits)	MPC	HE	MORE-MDL
100	82.76	85.43	89.12
200	84.72	87.28	90.68
300	86.09	88.15	92.56

The above table 2. represents the security strength results between the proposed and existing methods.

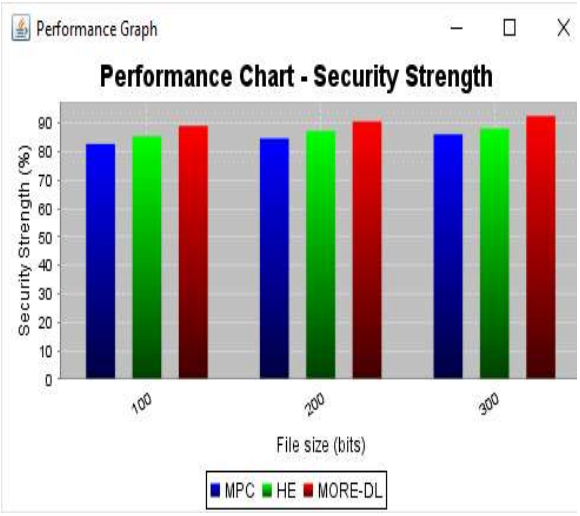

Figure 6. Performance comparison of security strength between the proposed MORE-MDL and existing methods

The figure 6. illustrate the performance comparison of security strength among proposed and present techniques. The linear nature of MORE is the most major security problem, as most encryption methods are founded on severely nonlinear functions and modular arithmetic over huge integers. By providing permission to big amount encrypted and unencrypted data pairs, this linearity might permit one to deduce secret key. As per the analysis, the proposed model has a higher security strength than present encryption technique.

Table 3. Comparison table for detection accuracy between the proposed and existing methods

File size (bits)	MPC	HE	MORE-MDL
100	82.55	85.29	87.15
200	84.76	87.43	89.76
300	91.87	93.57	94.34

The above table 3. defines the detection accuracy results between the proposed and existing methods.

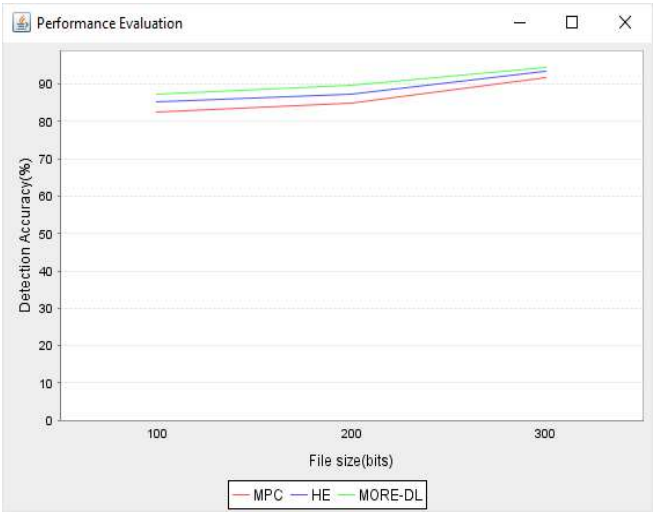

Figure 7. Performance of Detection Accuracy among proposed MORE-MDL with existing methods

The performance comparison of Detection Accuracy among proposed and current techniques is shown in Figure 7. Any MDNN model's performance may be enhanced in general by changing the network's design or using better activation functions along with optimization methods. Furthermore, training data augmentation methods, such as elastic distortions, is used to reduce classification error rate even more.

Table 4. Comparison table for Run time between the proposed and existing methods

File size (bits)	MPC	HE	MORE-MDL
100	0.68	0.61	0.57
200	0.77	0.72	0.69
300	0.85	0.80	0.76

The above table 4. represents the run time results between the proposed and existing methods.

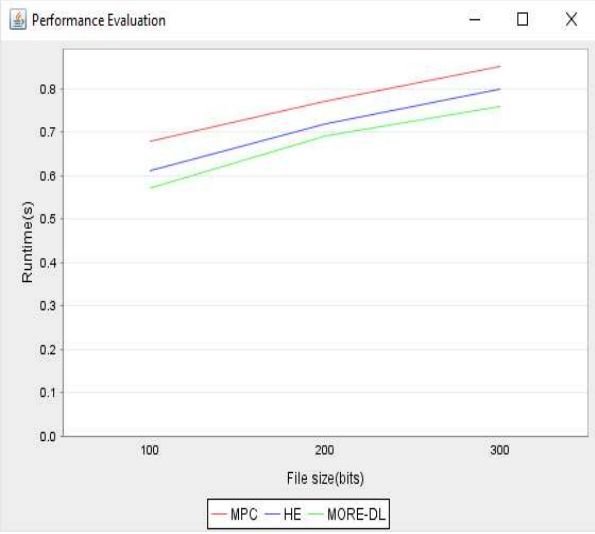

Figure .8. Runtime Comparison Between The Proposed MORE-MDL And Existing Methods

The figure 8 shows a comprehensive differentiation of the runtime for each works. Data was used to achieve all of the study findings, both during training and inference. Despite the fact that MDL methods execute right on MORE, homomorphically encrypted data is considerably slower throughput training and inference, system is currently exceptionally quicker contrasted with the conventional fully homomorphic encryption systems, with a difference of around 6 to 7 orders of magnitude, while carrying out fundamental algebraic process.

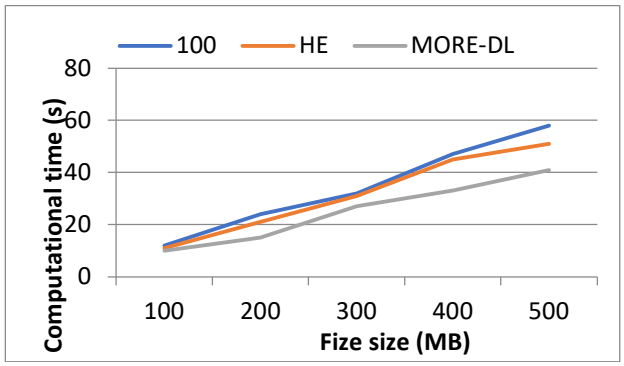

Figure 9. Time complexity analysis between the proposed and existing algorithms based on the size of the file.

The time complexity of each algorithm is shown in Fig. 9,where the vertical axis represents the computational time and the horizontal axis represents the size of the file. Fig. 9. indicate that the proposed algorithm has the best detection accuracy of all methods with a comparable order of time complexity. Compared with the proposed algorithm, the accuracy of the proposed model is high, and the time complexity is less compared to the existing methods. Thus, the method described in this work achieves a good accuracy and less time complexity.

5. CONCLUSION

In this paper, a deep learning system with significant privacy enhancements based on cryptographic primitives and distributed training data is introduced. MORE, a proposed privacy-preserving training technique based on encryption technique, allows calculations inside a MDNN system conducted on floating-point having quite low operational cost. For privacy-preserving computations within MDL methods, a variation of noise-free matrix-based homomorphic encryption techniques were developed. However, a homomorphic cryptosystem uses a private key to encrypt and decrypt information, it differs from other encryption types in that it retains arithmetical features of the encrypted data, allowing a diversity of processes made directly on encrypted data without necessitating access to decrypted data or key. After examining well-known MNIST digit identification issues in order to assess the viability of the proposed technique and demonstrate that performance does not suffer when MDL is used to MORE homomorphic data. The findings of the test demonstrate that the proposed MORE-MDL techniques have high classification accuracy, demonstrating the encryption's reliability. Further,

this proposed MORE-MDL based model provides the solutions to protect the confidentiality of sensitive health information, while encouraging the evolution of personalized medicine by protecting the integrity of patient health data. Further this work focus on the process of improving the classification performance using hybrid deep learning model.

ACKNOWLEDGEMENTS

The author[1] extends his sincere gratitude to Vasavi College of Engineering, Hyderabad and GITAM University for its support in this regard.

REFERENCES

[1] Butun, I., Kantarci, B., & Erol-Kantarci, M. (2015,). Anomaly detection and privacy preservation in cloud-centric internet of things. In *2015 IEEE International Conference on Communication Workshop (ICCW)* (pp. 2610-2615). Ieee.

[2] Vennila, S., & Priyadarshini, J. (2015). Scalable privacy preservation in big data a survey. *Procedia Computer Science*, *50*, 369-373.

[3] Zhang, K., Liang, X., Baura, M., Lu, R., & Shen, X. S. (2014). PHDA: A priority-based health data aggregation with privacy preservation for cloud assisted WBANs. *Information Sciences*, *284*, 130-141.

[4] Kaaniche, N., & Laurent, M. (2017). Data security and privacy preservation in cloud storage environments based on cryptographic mechanisms. *Computer Communications*, *111*, 120-141.

[5] Graepel, T., Lauter, K., &Naehrig, M. (2012,). ML confidential: Machine learning on encrypted data. In *International Conference on Information Security and Cryptology* (pp. 1-21). Springer, Berlin, Heidelberg.

[6] Zhang, K., Liang, X., Baura, M., Lu, R., & Shen, X. S. (2014). PHDA: A priority-based health data aggregation with privacy preservation for cloud assisted WBANs. *Information Sciences*, *284*, 130-141.

[7] Tang, J., Cui, Y., Li, Q., Ren, K., Liu, J., &Buyya, R. (2016). Ensuring security and privacy preservation for cloud data services. *ACM Computing Surveys (CSUR)*, *49*(1), 1-39.

[8] Zhang, X., Yang, C., Nepal, S., Liu, C., Dou, W., & Chen, J. (2013, September). A MapReduce based approach of scalable multidimensional anonymization for big data privacy preservation on cloud. In *2013 International conference on cloud and green computing* (pp. 105-112). IEEE.

[9] Hu, B., Murata, Y., & Murayama, J. (2015). Security information sharing platform over multiple services. In *2015 10th Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT)* (pp. 1-3). IEEE.

[10] Alamri, F. S., & Lee, K. D. (2015). Secure sharing of health data over cloud. In *2015 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW)* (pp. 1-5). IEEE.

[11] Raj, A., Arunprasath, R., &Vigneshwari, S. (2016, March). Efficient mechanism for sharing private data in a secured manner. In *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)* (pp. 1-4). IEEE.

[12] Shen, M., Tang, X., Zhu, L., Du, X., &Guizani, M. (2019). Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities. *IEEE Internet of Things Journal*, *6*(5), 7702-7712.

[13] Shortell, T., &Shokoufandeh, A. (2015, October). Secure signal processing using fully homomorphic encryption. In *International Conference on Advanced Concepts for Intelligent Vision Systems* (pp. 93-104). Springer, Cham.

[14] Chen, Y., & Gong, G. (2015, September). Integer arithmetic over ciphertext and homomorphic data aggregation. In *2015 IEEE Conference on Communications and Network Security (CNS)* (pp. 628-632). IEEE.

[15] Wang, Q., Zeng, W., & Tian, J. (2014, July). Compressive sensing based secure multiparty privacy preserving framework for collaborative data-mining and signal processing. In *2014 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1-6). IEEE.

[16] Rahulamathavan, Y., Phan, R. C. W., Veluru, S., Cumanan, K., &Rajarajan, M. (2013). Privacy-preserving multi-class support vector machine for outsourcing the data classification in cloud. *IEEE Transactions on Dependable and Secure Computing*, *11*(5), 467-479.

[17] Yu, C., Ding, Z., & Chen, X. (2021). HOPE: Software Defect Prediction Model Construction Method via Homomorphic Encryption. *IEEE Access*, *9*, 69405-69417.

[18] Kaaniche, N., & Laurent, M. (2017). Data security and privacy preservation in cloud

storage environments based on cryptographic mechanisms. *Computer Communications*, *111*, 120-141.

[19] Xu, K., Yue, H., Guo, L., Guo, Y., & Fang, Y. (2015). Privacy-preserving machine learning algorithms for big data systems. In *2015 IEEE 35th international conference on distributed computing systems* (pp. 318-327). IEEE.

[20] Fung, B. C., Wang, K., & Yu, P. S. (2005, April). Top-down specialization for information and privacy preservation. In *21st international conference on data engineering (ICDE'05)* (pp. 205-216). IEEE.

[21] Laur, S., Lipmaa, H., &Mielikäinen, T. (2006). Cryptographically private support vector machines. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 618-624).

[22] Li, J., Kuang, X., Lin, S., Ma, X., & Tang, Y. (2020). Privacy preservation for machine learning training and classification based on homomorphic encryption schemes. *Information Sciences*, *526*, 166-179.

[23] http://yann.lecun.com/exdb/mnist/.

[24] P. Patro, K. Kumar, G. S. Kumar, G. Swain, Similarity and wavelet transform based data partitioning and parameter learning for fuzzy neural network, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 6, Part B, 2022, Pages 3424-3432, ISSN 1319-1578.

[25] Kathiresan, V., & Sumathi, P. (2012). An efficient clustering algorithm based on Z-score ranking method. In *2012 International Conference on Computer Communication and Informatics* (pp. 1-4). IEEE.

[26] P. Patro, K. Kumar, & G. Suresh Kumar(2020). Neuro Fuzzy System with Hybrid Ant Colony Particle Swarm Optimization (HASO) and Robust Activation, Jour of Adv Research in Dynamical & Control Systems, Vol. 12, 03-Special Issue (pp. 741-750).

[27] Gupta, C. P., & Sharma, I. (2013). A fully homomorphic encryption scheme with symmetric keys with application to private data processing in clouds. In *2013 Fourth International Conference on the Network of the Future (NoF)* (pp. 1-4). IEEE.

[28] P. Patro, K. Kumar, & G. Suresh Kumar (2020). Optimized Hybridization of Ant Colony Optimization and Genetic Algorithm (HACOGA) Based IC-FNN Classifier for Abalone. Journal of Computational and Theoretical Nanoscience Vol. 17, pp.2756–2764.

[29] Chen, Y., Lin, Z., Zhao, X., Wang, G., & Gu, Y. (2014). Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected topics in applied earth observations and remote sensing*, *7*(6), 2094-2107.

[30] Ishibuchi, H., & Nii, M. (2001). Numerical analysis of the learning of fuzzified neural networks from fuzzy if–then rules. *Fuzzy sets and Systems*, *120*(2), 281-307.

[31] Gandomi, A. H., Yang, X. S., & Alavi, A. H. (2013). Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with computers*, *29*(1), 17-35.

[32] Ogiela, L.; Ogiela, M.R.; Ko, H. (2020). Intelligent Data Management and Security in Cloud Computing. Sensors, 20,3458. https://doi.org/10.3390/s20123458.

[33] M. Mayuranathan, S.K. Saravanan, B. Muthusenthil, A. Samydurai (2022) An efficient optimal security system for intrusion detection in cloud computing environment using hybrid deep learning technique, Advances in Engineering Software, Volume 173, 103236,ISSN0965-9978, https://doi.org/10.1016/j.advengsoft.2022.103236.