IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 14, 2021, accepted October 17, 2021, date of publication October 26, 2021, date of current version November 2, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3122938*

# Enhanced Security in Cloud Computing Using Neural Network and Encryption

**MUHAMMAD USMAN SANA[1], ZHANLI LI[1], FAWAD JAVAID[2], HANNAN BIN LIAQAT[3], AND MUHAMMAD USMAN ALI[4]**

[1]College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an, Shaanxi 710054, China
[2]Department of Communication and Information Engineering, Xi'an University of Science and Technology, Xi'an, Shaanxi 710054, China
[3]Department of Information Technology, University of Gujrat, Gujrat, Punjab 50700, Pakistan
[4]Department of Computer Science, University of Gujrat, Gujrat, Punjab 50700, Pakistan

Corresponding author: Muhammad Usman Sana (m.usman@uog.edu.pk)

This work was supported by the Natural Science Basic Research Plan in Shaanxi Province of China under Program 2019JM-348.

**ABSTRACT** With the fast advancement in cloud computing, progressively more users store their applications and data on the cloud. Cloud computing has lots of features, e.g. virtualization, multi-user, efficiency, cost savings, and most importantly security. Machine learning approaches based on neural networks are being widely applied in cloud infrastructure when training is performed however this may produce possible privacy and security risk as direct access to raw data is required. To address this problem, we propose a new security design using Artificial Neural Networks (ANN) and encryption to confirm a safe communication system in the cloud environment, by letting the third parties access the data in an encrypted form for processing without disclosing the data of the provider party to secure important information. In this paper, to train neural networks using encrypted data we considered the Matrix Operation-based Randomization and Encipherment (MORE) technique, based on Fully Homomorphic Encryption (FHE). This technique allows the computations to be performed directly on floating-point data within a neural network with a minor computational overhead. We examined the speech and voice recognition problem and the performance of the proposed method has been validated in MATLAB simulation. Results showed that applying neural network training with MORE offers improved accuracy, runtime, and performance. These results highlight the potential of the proposed method to protect privacy and provide high accuracy in a reasonable amount of time when compared to other state-of-the-art techniques.

**INDEX TERMS** Ciphertext, cloud computing, homomorphic encryption, matrix operation-based randomization and encipherment, neural network.

## I. INTRODUCTION

The In the past decade, demand for cloud computing among businesses and individual users is increasing immensely because of numerous reasons including, improved productivity, efficiency and speed, cost savings, performance, and most importantly security. The current advancement in cloud computing has significantly changed everyone's perspective on development models, software delivery, and infrastructure architectures. Some of the main advantages of using a cloud environment are the availability of resources, cost reduction, and storage flexibility. Despite these gains cloud computing also has certain issues, and the most important one is security. Cloud adopts all security problems of its modules because of its complex framework. In cloud security, a relatively novel approach is Artificial Neural Networks (ANN). ANN technique in cloud is applied mainly in the detection of intrusion in security [1]. Machine learning is a subdivision of artificial intelligence. Software-as-a-service providers are applying machine learning tools to bigger software groups to provide greater productivity and functionality to end-users [2]. The brain is an ideal example that basically learns with experiences. The neural structure of the brain is the inspiration behind the ANN. After the natural procedure of thinking has been studied in biological research, it was revealed that the brain saves information in the form of complex patterns. A new research field in computer science is arises based on the methods which result in keeping the data

The associate editor coordinating the review of this manuscript and approving it for publication was Jose Saldana.

as patterns. Nowadays large parallel networks are required to be created which then are trained when a particular problem needs to be solved [3].

Recent developments in neural networks have attained amazing success in a variety of applications, including, language representation, image classification, etc. This progress is enabled with the accessibility of large and representative datasets to be trained by neural networks. These datasets are usually from several sources and may comprise of sensitive information. This requires a technology that meets the needs of an application which provides good and strict privacy assurance. In the past machine learning methods and advanced privacy mechanisms are combined to train neural networks with an affordable security budget. This may achieve good results in convex models with smaller parameters or dealing with non-convex targets, multiple layers, and models with tens to millions of parameters but in complex neural networks, they can only be processed with huge loss of privacy [4]. With strong computational supremacy and the ability to resolve large datasets, Artificial Neural Networks in machine learning is the most widely used model and is continuing to evolve [5].

Homomorphic Encryption (HE) is a method that without compromising the encryption process enables to make computations over encoded data. Traditional encryption systems rely on sharing keys between peers involved in swapping encoded messages. However, these techniques do raise confidentiality concerns. The user or deal provider that owns the key has limited rights to the data. Especially when using the general cloud service station, the user loses control over the secrecy of important data. If the key is not public, the encoded information will be public with third parties but the owner does not want them to access the details. In addition, members of staff serving, suppliers, and workers will be able to retain the selected user items for a long time even after the users cut their relationship with untrusted cloud services. HE is a proprietary encrypted pattern that can solve this problem as it permits the third parties to work with the encoded data without decoding the encrypted data first. Although this useful feature of HE has been around for more than a decade, Craig Gentry launched the first fully-achievable encryption scheme known as Fully Homomorphic Encryption (FHE) in 2009 to be implemented on encoded data. This is a huge achievement, but there are other advanced programs so far that indicate that FHE still needs major improvements to become applicable for all platforms [6]. The main cause of the incompetence of the FHE scheme is the reality that these encryption schemes are based on matrix problems that are integrally blaring: for example, encrypting the integer message and then decryption is applied with a minor error in this give an alarming message for security purposes in the encryption process. When applying only encryption and decryption processes this is not much of a concern because by scaling the message with suitable element the error can be removed without difficulty [7].

In recent years, great effort has been made to develop a variety of privacy protection techniques that can bridge the gap between utility and data privacy. Among these approaches, numerous encryption techniques are starting to grow rapidly, including HE that secures multi-terminal computing, and gap privacy. This technology guarantees data privacy while allowing data to be outsourced to commercialized cloud computing systems for processing, all despite the fact that the data is encrypted. This technology shows encouraging results, but their use in modern machine learning applications is still limited since it relies heavily on specialized and dedicated user-server applications to achieve accurate function. Also, among these technologies, there is always an exchange between privacy and performance, as each technology has certain advantages and disadvantages

Much has been accomplished in Neural Networks in recent years. It has succeeded in speech recognition, natural language processing, and image classification. Several efforts have also been made to solve the issue of data privacy in deep learning. In this paper we applied MORE scheme because it perform random set of techniques directly on the encoded information without disclosing basic data or encryption keys. This feature is especially useful in the perspective of deep learning because it guarantees that the data confidentiality and forecasts are preserved while the data is being processed.

In this paper, we proposed an improved type of encoding technique that allows us to encrypt data as it is being processed. Therefore, we aim to maintain the confidentiality of the data by permitting third parties to access the information in an encrypted form without disclosing the basic data. We explain the following research questions through our results:

- How do we use HE for data alteration in the cloud?
- How to Encode Plain Text with MORE scheme?
- What is the ANN process of enhancing security?
- What are the security models included in ANN over encryption?

The contribution of this research is an understanding of how deep learning methods and tools help to create a secure security system with encryption of user data. To transfer sensitive data without violating confidentiality, we need to encrypt the data anonymously. The use of anonymous encrypted data limits the ability of neural networks to elicit valuable information and insights from that data. In this research, we examine how we use a method based on Homomorphic Encoding to address the limitations and to maintain confidentiality. We find the solution that how to perform neural network prediction on encoded data as it is a non-linear model of machine learning with big modular capability and also explain how the proposed method improving the efficiency of the cryptographic model in real-world applications. The key objective of this study is to formed a cloud-based ''safe'' structure for storing data on a cloud platform, to spread new security ideas in the future for cloud setup and data migration in the virtual world. The biggest problem with implementing this solution is the

fact that the activation function commonly used in neural networks is not polynomial. They contain linear sigmoidal functions. In this paper, we provide detailed information about the neural network model developed for the application under study, also show their runtime performance and model accuracy results.

## II. LITERATURE REVIEW

Schemes based on HE in neural networks have been suggested to solve the problem of nonlinear activation functions by establishing a shared protocol between data holders and model vendors. In short, every nonlinear transformation is estimated by the data owner. The model sends input to the nonlinear transformation in encrypted form, then decrypts the data of the owner, applies the conversion, encrypts the result, and sends it again. Unfortunately, these interactions involve a lot of response time and add difficulties for the owner of the data, which is arbitrary. Also, information about the model is leaked. Consequently, to alleviate this problem, security mechanisms such as arbitrary execution orders have been introduced. On the contrary, the presented process does not require complex connection diagrams. The data is encrypted and transmitted to the owner of the data. The model does the calculation and sends the prediction (coding) again [8].

A homogeneous cryptographic technology based on estimations is proposed in Asiacrypt 2017, which is both theoretically efficient and highly practical. Complete key recovery is possible with high probability and very small execution time. This technique is implemented and tested attacks against major homogeneous open source cryptographic libraries including HEAAN, SEAL, and PALISADE, as well as against a number of features often seen in machine learning of encoded data using CKKS diagrams such as average and variance calculations, Maclaurin series is access for logical and exponential functions. This attack cannot obtain complete security against passive enemies when the current expression of IND-CPA security is executed by CKKS (or indistinguishable from the chosen-plaintext attacks) is applied to a similar cryptographic scheme, and to assess its integrity. It indicates the need for a stronger plan for security [7].

For a wide range of applications, numerous corporations offer neural network prediction services to customers. Though, one party's privacy is compromised by using current prediction schemes: either the provider of services must store on the customer's device its exclusive neural networks or the customer has to show secretive inputs to the provider for services. Both ways are not appropriate because this will disclose the service supplier's exclusive model and also expose the private secretive information of the customer. Authors [9] implement, estimate, and design, a secure prediction scheme DELPHI that allows performing neural network inference between two parties without disclosing the data of both parties. By instantaneously co-designing machine learning and cryptography DELPHI solved the problem. The authors proposed a hybrid cryptographic procedure that improves upon the computation and communication costs as compared to previous work. They also provide developer neural network structure configurations that show the improved accuracy and performance of this proposed hybrid procedure.

Chameleon, an innovative hybrid (mixed-protocol) architecture was presented [10] for secure function evaluation that without revealing their private inputs allows two parties to mutually compute a function. Chameleon combines additive secret sharing with the finest characteristics of standard secure function evaluation procedures. Specifically, the architecture implements linear operations by using additively secret shared values in the ring and using the Goldreich-Micali-Wigderson and Yao's Garbled Circuits protocols for nonlinear operations.

In the cloud when information is transferred over the internet, retaining its security is the main issue. Therefore, data stored also needs security employing standard encryption methods in the cloud. On the contrary, in cryptographic systems, the receiver or the second party to decrypt the data needs to have the sender's private key. Hence, every single time in the cloud when a user sends a requisition to its virtual atmosphere and assumes secure and fast computing on its information, it must provide a private key along with the request of the user, and the processing is completed after decoding of the data. Though each time it is computed, the threat of the key being disclosed will enhance. In such a situation users will have to alter the key, and in case of the key being leaked they need to reproduce the secret key over again and if symmetric encryption is being applied, both parties are required to have a similar secret key. Regardless of the great speed, security violations will enhance computational overheads and processing time. Privacy-preserving mechanisms can resolve this concern. The technique and comprehensive solution that preserves data privacy is encryption [11].

Maintaining security and data privacy does not involve only careful attention but also needs precise predictions when applying machine learning to an issue that includes financial, medical, or sensitive data. Ethical and legal boundaries may stop the use of machine learning solutions on the cloud for such jobs [12]. Proposed a technique in which they transform learned neural networks to CryptoNets. This permits the owner of data to send their data to hosts of the network which is a cloud service in an encrypted form. The data remains confidential because of the encryption and the cloud does not have access to the keys required to decrypt the data. However, to make encoded predictions to the encoded data the cloud service can use the neural network and give them back data in encoded form. The owner of the secret key can take back these encoded predictions and can also decode them. Hence, the cloud facility does not attain any data about the prediction it prepared or about the raw data. The MNIST optical character recognition jobs determine on CryptoNets. CryptoNets can create about 59000 predictions per hour on a single PC and attains 99% accuracy. Thus, agree to, private and accurate predictions and high throughput.

Recent signs of progress in machine learning have increased the range of Neural Network (NN) interpretation in

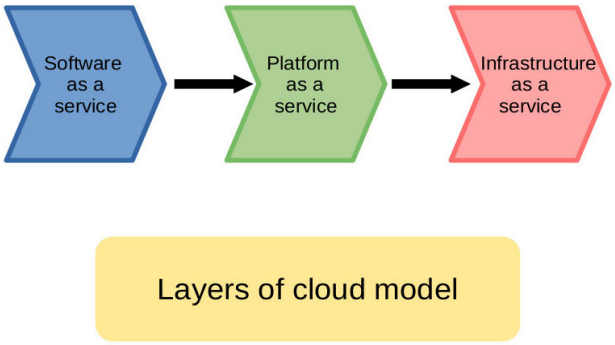



**FIGURE 1.** Layer of cloud computing.

general uses such as recognization of voice and classification of the image [13], [14]. Provide a well-organized structure for enhancing the security of Cloud Computing Internet of things based on intelligent transportation systems. The suggested structure allows without disclosing any sensitive data extracting specific vehicle information. In the direction of this objective, the author applies a convolutional neural network to categorize encrypted images, in real-time based on vehicle type, achieved with cameras incorporated into units on road-side that based on intelligent transportation system making hidden all the sensitive information. Within the suggested outline, the Authors improve a novel image classification design that protects the personal information of drivers' by not completely decrypting the images, such as license plate, location, and vehicle contents. In addition, as compared to conventional systems the structure does not want an entirely decrypted image that increases the computational effectiveness of the system. The achieved results demonstrate that the suggested partially decryption classification method shows up to an 18% decrease in complication in computational work when matched with the system which is fully decrypted. The basic idea of the projected collection of methods is to consider and associate numerous different works and apply the common elective judgment method to get a more well-organized classification for each work. The recreation consequences demonstrate that the projected technique greatly reduces the time complexity and retains high accuracy compared to other existing techniques.

## III. FUNDAMENTALS OF THE CLOUD COMPUTING SYSTEM

Generally, a structural design of a cloud computing system consists of three levels. Characteristic levels of cloud computing services are shown in Fig. 1.

- Platform as A Service
- Software as A Service
- Infrastructure as A Service

### A. PLATFORM AS A SERVICE (PaaS)

With this type of service, the software can be developed at no cost, provide hassle-free purchasing and handling of main hardware and software, and also provide virtualized hosting services. Software developers can construct novel applications or upgrade longstanding applications effortlessly. They do not have to bear the cost of application development. This service will offer a level of system software that can use and create an advanced level of service. The platform services include centralized tools, integration tools, messaging and information exchange, and communication setup. For example, applications created on the Google Apps engine are wholly retained by Google, and cannot give clients the services that it cannot provide in that package. Many corporations have industrialized stages that permit end users to route their applications over a central server over the Internet. Examples of these services are Microsoft's Azure OS and Google Apps [15].

### B. SOFTWARE AS A SERVICE (SaaS)

It is an application service or cloud-as-a-service that can provide the software an online service, which eliminates the need for software installation on client computers and makes repair and sustenance easy. The key characteristics of this service are:

- Network communication and administration of business software. Actions are controlled by particular centers and centers are located someplace else other than the location of every single client, allowing clients to get requests slightly over the network. Compared to a one-to-one model, the service delivery model is nearer to a one-to-many model.
- Centrally manage software upgrades and without the requirement of upgrades or downloading patches. Including security services, for example, MessageLabs P2P software, for example, Skype; software services, for example, Google Labs, salesforce.com, CRM, HR, IBM Lotus, Payroll, Gmail, Google Calendar, and Live web applications, for example, YouTube Twitter, and Facebook software storage services, add-on services, for example, Microsoft Online Services; are examples of application layers. The aforementioned companies were created specifically for Software as a Service. When customers get the application over the internet network, these companies charge a user registration fee and install the software on a central server [16].

### C. INFRASTRUCTURE AS A SERVICE (IaaS)

It gives the computer infrastructure commonly an effective platform as a service and is known as a cloud infrastructure service. Users do not need to buy software and hardware, network tools, or space for data centers but they can purchase this infrastructure entirely as a full subcontracted service. Cost of Services is often charged according to the resources used and based on the computing model they utilized therefore charges be determined by the amount of usage. In fact, this technique is a development of the private virtual service delivery model, which usually has a virtual computing framework. Generally, this computer infrastructure service and virtualization framework can be provided as a service. One such example is Amazon AWS services [17].

## IV. HOMOMORPHIC ENCRYPTION

A fully homomorphic encryption structure is first introduced [18], In the review [19] many modifications in the original scheme were recommended. Many of these strategies are computationally intensive and in terms of security recognized for their efficiency. Decipherment is no longer possible only a restricted number of processes can be implemented. In real-world applications, this noticeably limited their usability. Features like computations take part in numerous levels of magnitude slower than the plaintext corresponding parts gathered noise which restricted all computations being applied in modulo N and the total number of processes that can be completed, for the collaboration of data analysis and deep learning this act a big barrier. Presently no existing strategies can handle rational numbers whereas improvements in HE led to modifications of encryption techniques.

In recent years numerous open-source HE techniques have developed, based on the involved encryption strategy each one with different characteristics [20]. Simple Encrypted Arithmetic Library (SEAL) of Microsoft strategy [21], with sustenance for the Cheon-Kim-Kim-Song (CKKS) strategy [22], the Brakerski/Fan-Vercauteren strategy Fan and Vercauteren (2012) based on the Brakerski-Gentry-Vaikuntanathan strategy and HELib of IBM [23] are two most extensive used HE techniques. The absence of support for floating-point numbers is an observable limitation of HELib is SEAL takes benefit of a specific property of the CKKS strategy, computation allows to be implemented on rational numbers, without altering the encrypted value, rescaling can be implemented. Floating-point characteristics of the information are scaled by a factor that disturbs the accuracy of the computation as plaintext and is characterized by using integer coefficients as a polynomial. Homomorphic processes are done with both SEAL and HELib lead to noise, therefore restricting the total processes that can be done by means of ciphertexts. To retain the noise level under a certain margin, noise-management methods have been incorporated, to the extent that the ciphertext is not turn out to be degraded. Though SEAL practices a scale-invariant error decreasing procedure that will be completed as theoretical information which needs a number of operations estimation. HELib uses the costly technique of bootstrapping to allow unrestricted computations. Furthermore, based on the kinds of procedures that are implemented on the ciphertext there will be some limitations. The strategies used in SEAL and HELib concerning multiplication and addition are fully homomorphic and merely polynomial functions can be simply implemented. Consequently, there is no hidden backing for nonlinear functions and division, and by this low-degree polynomials are estimated.

Although in terms of proven security these strategies are identified for their efficacy, the aforementioned limitations, along with the computational expenses, in the neural network topology present clear restraints, which as a result upset privacy-maintaining neural networks performance [24].

Instead of FHE other recommended approaches depend on partially homomorphic encryption (PHE) also implemented. Presently in a real-world system, FHE is practically difficult to be used so a practical method in a system that is specified only for exact processes based on PHE will be used. This technique may be used in a real-world application with acceptable overhead and presents a clear benefit in terms of running time.

Algebra homomorphic encryption strategy is also very encouraging [25] this is an encryption scheme in which both multiplications and addition can be implemented on encoded data that is homomorphic regarding algebraic multiplication and addition. A comparatively lesser computational complication is the main benefit of this strategy, like ElGamal and Paillier, however, being homomorphic with multiplication and addition. This scheme only permits the encryption of comparatively small integer numbers and this is the key limitation of this approach along with ElGamal and Paillier. More precisely, an exponentiation process requires to be assessed, during the encryption process, where the message to be encoded is the exponent. Therefore, even with a multi-accurate algebraic library, the process even though creates an overflow. With 1024-bit integers, it was found only the largest number that can be encoded is up to about $10^3$. These limitations become extra significant when accomplishing mathematical operations on encoded data. In other words, cannot fix if the amount of encryption is too large to perform certain operations.

To ease privacy-based Deep Learning analysis, the cryptographic system should be capable to perform computations on model rational information. To meet this requirement, cryptographic mechanisms are usually used to encrypt a specific rational digit into a series of integers. Few fundamental operations are tough to apply on encrypted data, and when applied to real data, this method has limited functionality. Additionally, the markup scheme not only bounds data usage but also openly disturbs the computation outcomes [26]. Secure multi-party computation (SMPC) technology offers a promising data privacy solution by allowing the analysis of complex data to be scattered among different information suppliers and not to reveal sensitive data outside the results of the analysis. Though the aim of SMPC is not novel, recent technical and hardware advancements have led to more ways to use SMPC in this field to ensure data privacy in machine learning applications. The first challenge was to train the NN model in an SMPC setup, in which the NN-based analysis was performed by underground distribution, unintended transmission, and chaotic circuits in the safe bidirectional computation of logical networks. The biggest problem with SMPC for machine learning is the calculation of non-linear functions. This is because these processes cause a spike in training time. Also, availability is becoming increasingly limited by the time required to communicate. Although there are investments underway to improve the technology, SMPC quiet requires a lot of communication, which is not possible in machine learning, which requires a lot of data. Increasing

the number of participants or the complexity of the model has a significant impact on the cost of communication and calculation [27].

Cloud computing is wide-open to huge internal and external privacy leakage and breaches threats. In the cloud, for big data authors present a privacy-preserving distributed analytics structure. FHE is used as a developing and controlling cryptosystem that on encrypted data can implement analysis tasks. In cloud computing to partition equally data and analysis of computations into subset nodes that can be implemented autonomously. The recognized distributed technique has scalability. This quickly speeds up the performance of encoded data processing whereas preserving a great level of accuracy of the analysis. Evaluation of experimental cloud-enabled applications for building secure analytics according to both accuracy and performance analysis regulates the efficiency of the planned framework [28]. Authors using HE technique, which without decrypting ciphertexts allows cloud computing environment to execute arithmetic operations. By means of the HE scheme, consumers can be able to provide only ciphertexts for using Reinforcement learning-based services to the platform of cloud computing. A privacy-preserving reinforcement learning structure for the platform of cloud. Focused on learning with errors the proposed framework exploits a cryptosystem for FHE. Estimation and analysis of performance for the proposed privacy-preserving reinforcement learning framework are observed in multiple intelligent service scenarios based on cloud computing [29].

With the advancement in cloud computing, techniques of data analysis play a significant role to provide massive market values. To perform certain linked mining tasks clients with limited resources in computing may use the option of the cloud. In this process, data owners may have a possibility of private information leakage of sensitive information. Owners of data may encode raw data prior to uploading to reestablish privacy in outsourced data. In recent years analysis of encoded data is a daring task, the attention of numerous researchers is attracted towards this area. A cryptographic tool is required to solve this challenge which is HE. It enables the processing of data without decryption of encrypted data. Investigating HE arrangements in a multi-key environment that keep privacy-preserving data mining has become a significant way. A unique homomorphic cryptosystem, which manages numerous cloud users to have altered public keys is proposed. Moreover, show that our technology is practically achievable on a real transaction database [30].

Over non-abelian rings, a new HE scheme is proposed, and in ciphertexts space homomorphism operations are defined. One-way security can achieve by the scheme. Over a matrix-ring, HE is proposed. Established on the homomorphism of two order displacement matrix coding function supports encryption of real numbers and attains fast homomorphic evaluation of ciphertexts without the decryption of any ciphertexts operations and transitional outcome. Moreover, in the data ciphertexts environment for training in machine learning and classification, the scheme realizes privacy preservation. The investigation shows that the proposed technique is effective for homomorphic operations and encryption/decryption [31].

In recent years, many HE approaches have been established to encounter cloud security requests. Though this method is very safe, most of the techniques are poorly performed because it is heavily subjective to processing time, which is slower than plain text computations. This limits its availability in practical applications. Hence, simplified coding strategies based on linear transformation are emerging as a more practical and viable alternative in this area [32], [33].

To preserve privacy in real-world up-to-date applications in the cloud exceptional strategy of a HE algorithm is required that permits the computation over the data which is encrypted. For applications in the real world, the current solutions are not practical. Symmetric methods undergo little immunity in case of attacks for example known and chosen plaintext attacks. Whereas Asymmetric methods suffer from great overhead computation. Authors [34] are building a new algorithm that overcomes the shortcomings of MORE by focusing on the MORE method simply considering the symmetric method. They describe and evaluate the proposed algorithm in detail. The safety enactment consequences indicate that the proposed method can prevent attacks on security and the analysis of performance presents that the suggested method can avoid the strong attacks without deprivation of the performances of the system in terms of consuming energy and latency.

Despite researchers criticizing it for poor security still this type of cryptographic system presently appears to be the only way to implement confidentiality protective computing in real-world applications. Consequently, in this paper, we applied this technique depending on a variation of the matrix-operation-based homomorphic encoding. As compared to the existing accepted techniques in neural networks based on privacy-preserving solutions [12], [35], [36] the MORE encoding strategy is non-deterministic and noise-free. On ciphertext data, a limitless number of processes can be implemented. Same plaintext data with multiple encryptions, outcome shows different ciphertexts with the same key. A general algorithm for converting text into the encrypted and unencrypted form is shown in Table 1. In addition, the MORE strategy can perform principle arithmetic operations on the encoded data. In this paper, MORE has been updated to responsively support floating-point accuracy for data analysis based on neural networks for privacy.

## V. MATRIX-BASED OPERATION FOR DATA RANDOMIZATION AND ENCIPHERMENT

A MORE coding system variant, which is suitable for working directly on floating-point data, was considered. According to the MORE coding strategy, the standard number of plain text is encoded into an nxn ciphertext array and the matrix algebra is used to compute the ciphertext data. Thus, all procedures are accomplished on the ciphertext information are known as Matrix-Operations. For example, the standard text development is expressed as ciphertext

**TABLE 1.** Algorithm convert ciphertext.

```
Algorithm 1:
Formula to convert data:
c=(x-n) %26   where n is key, 26 alphabets and, x is asci key plain text
#create function
i.    def encrypted (string, shift):
ii.   cipher=''      #currently no values or data available
        a.  for Char in String
                i.  if Char==' ':
                ii. Cipher=Cipher + Char
        b.  Else if Char. Isupper ():
                i.  Cipher= Cipher + Chr (ord(Char) + Shift - 65) % 26 + 65)
                ii.  #convert the uppercase alphabets
        c.  Else:
                i.  Cipher=Cipher + Chr (ord(Char) + Shift -97) % 26 + 97)    # #convert the lowercase alphabets
iii.  return cipher
iv.   #main window
v.    text=input ("enter the key")                #enter text you want to encrypt
vi.   s=int (input ("enter the shift key"))       #shift key use to add how many key you want to shift e.g. enter 2
vii.  print ("the original string is:", text)
viii. print ("the encrypted mgs is:", encrypted (text, s))
```

**TABLE 2.** More encoded structure.

| Communication | Scalar values |
|---|---|
| **Hidden key building** | Invertible matrix s belongs N $_{2x2}$ |
| **Matrix building** | M = m x r where these are random parameters |
| **Encrypted operation** | $M = V = SMS^{-1}$ |
| **Decrypted operation** | $V = L = S^{-1} VS$ |
| **Recover the text or message** | $m = L_{(1,1)}$ |

Matrix-Multiplication. In a Matrix Operation-based Randomization and Encipherment idea called MORE is considered for constructing a FHE scheme [37], [38] The arrangement of the arrays used to encode messages is an important factor in controlling the interchange between security and efficiency. For a 2 × 2 setting, MORE coding arrangements are shown in Table 2. The suggested technique is explained by the following Eq. (1):

$$M(m, k) = S^{-1} \begin{bmatrix} m & 0 \\ 0 & r \end{bmatrix} S \tag{1}$$

where r is a random integer, m the plaintext in a ring N, $S^{-1}$ its corresponding inverse and S is an invertible matrix in N (2 × 2) one.

The decryption method is purely the opposite of the encryption method by implementing as in Eq. (2):

$$V(m, k) = SM(m, k)S^{-1} = \begin{bmatrix} m & 0 \\ 0 & r \end{bmatrix} \tag{2}$$

The MORE approach is FHE one can see that since it fulfills all homomorphic properties. Nonetheless, this method offers extraordinary storage overhead and in [32] on MORE a significant recovery attack that needs only side-channel data on the plaintext. Thus providing an efficient FHE algorithm.

### A. ENCRYPTION OF COHERENT DATA

Like the FHE or PHE method, the novel MORE technique applies only to the N unit of positive integers, and all operations are performed by the N unit. To perform operations on rational numbers, this method is greatly based on an encryption mechanism. As a result, the real numbers are converted into integers or groups of integers, and then the numbers are encoded using this homomorphic scheme. A special way to develop code is to use continuous fraction points. Now we can get an accurate representation, but it is difficult to perform simple procedures on the numbers stated in this manner. Instead of this, by multiplying the coherent number to a big scaling element simpler encryption can be performed, for example, division, where this element is decreased. Or, enter a simpler symbol by increasing the sensible number by a large influence. It's smooth but needs a mechanism to regulate the scaling factor as for some tasks it is tough to accomplish in which division factors decrease the scaling factors that are difficult to implement must be modified. One of the basic advantages of this coding system is that it's framed for rational numbers. The disadvantage of this technique is susceptible to famous cryptographic attacks.

### B. IMPLEMENTING OPERATIONS ON ENCRYPTED INFORMATION

The MORE coding scheme has been found to be quite homomorphic for simple algebraic procedures. In real applications, including deep learning-based methods, must deal with non-linear functions. Many of the traditional methods used in non-linear actions are established on the basis of using a finite series of polynomials to arrive at a specific function. According to this method, the calculation of the nonlinear function is totally based on the algebraic operation, which is fully consistent with the MORE coding setting. However, it is easier to use this method in a system of MORE code words. Knowing that the predominant nature of the coding scheme and cipher-text procedures are dependent on Matrix-Algebra, non-linear parts can be calculated openly using Matrix-Functions.

## VI. NEURAL NETWORK

A neural network at an advanced level can be known as a computational model that over an arrangement of layers depict inputs to outputs with interrelated processing units (activation functions and transformations). Fig. 2 showed the structural design of a simple neural network, a non-linear activation function is added to each processing unit to create a mapping of complex arbitrary functions. It filters the data that passes over the network to determine the relevant input signal that will be passed to the next layer. It basically decides whether or not a particular neuron will be activated. In the absence of specific neurons, the NN develops as a plain linear model.

In machine learning using supervised machine learning, the model is designed to automatically recognize the

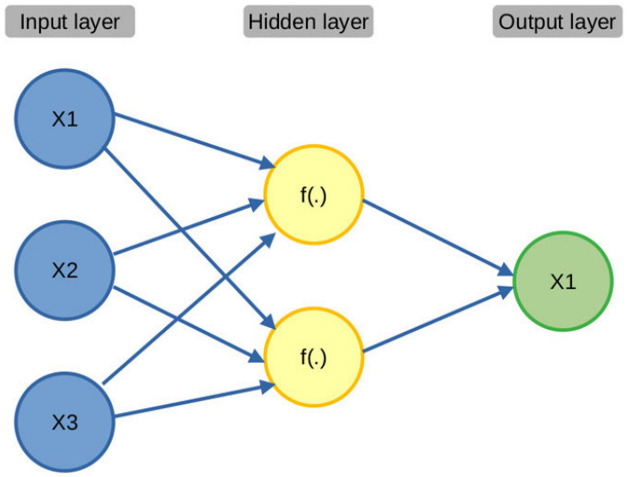



**FIGURE 2.** Design of neural network.

**TABLE 3.** Training network.

| *Algorithm 2:* | |
|---|---|
| 1. | *Procedure train (a, b)* |
| 2. | *Initialized parameters:* $\Theta_{h+1}$ |
| 3. | *for iteration Ɛ to M do* |
| 4. | *For set Ɛ to (N / y) do* |
| 5. | *Model a set of y train model:* $(a_x, b_y) \leftarrow \{a_i, b_i\}^{N}{}_{I=1}$ |
| 6. | *Calculate the prediction* |
| 7. | *Calculate loss* |
| 8. | *Calculate gradient of the loss with their output parameter* |
| 9. | *For L belongs to h do* |
| 10. | *Repeat step 8* |
| 11. | *End for* |
| 12. | *For I Ɛ h + 1 do* |
| 13. | *Keep informed the model:* $\Theta_i = \Theta_i$ *- T lambda* $\Theta_i$ |
| 14. | *end for* |
| 15. | *end for* |
| 16. | *end for* |
| 17. | *end process* |

mapping function based on repeated labeled training, reducing the fault function (such as loss) among the predictable and accomplished outputs. Table 3 Algorithm 2 enlightens the distinctive processes that are included in the training phase of the neural network. The complete training dataset whole processing is referred to as the epoch. For large datasets because of computational reasons, the parameter adjustment and later processing are completed in batches (subsets of data). The forward propagation stage delivers the expected outcome for the input samples providing a set of initialized parameters randomly in the first iteration. After that, the fault in function (loss) among the preferred output and predicted ones will be computed and distributed backward to reduce the total prediction error to conclude the direction through the network where every parameter has to be attuned. Lastly, to update the parameters of the network the achieved gradients are utilized by following a gradient descent (for example numerical optimization process). This procedure is repetitive over many iterations till the error in the network ends reducing

**TABLE 4.** More strategy.

| **Algorithm for Encryption:** | |
|---|---|
| 1. | ***Input: secret key s Ɛ N2*2*** |
| 2. | ***Input****: plaintext data m Ɛ N* |
| 3. | ***Output****: ciphertext V Ɛ* $N^{2*2}$ |
| 4. | *Fun Encrypt (m, S)* |
| 5. | *m Ɛ* $N^{2*2} \leftarrow$ *zero matrix* |
| 6. | *M (0, 0)* $\leftarrow m$ |
| 7. | *M (1, 1)* $\leftarrow$ *random variable (max value, min value)* |
| 8. | $V \leftarrow S^* M^* S^{-1}$ |
| 9. | *Return V* |
| 10. | *End fun* |

| **Algorithm for Decryption:** | |
|---|---|
| 1. | ***Input: secret key s Ɛ N2*2*** |
| 2. | ***Input****: ciphertext c Ɛ* $N^{2*2}$ |
| 3. | ***Output****: plaintext data m Ɛ N* |
| 4. | *Fun decrypt (V, S):* |
| 5. | $L \leftarrow S^{-1} * C^* S$ |
| 6. | $N \leftarrow L$ *(0, 0)* |
| 7. | *Return m* |
| 8. | *End fun* |

| **Algorithm for Key Generation:** | |
|---|---|
| 1. | ***Output****: secret key S Ɛ* $N^{2*2}$ |
| 2. | *Key generation ()* |
| 3. | *While true do* |
| 4. | *S* $\leftarrow$*random (size= (2, 2), max value, min value,)* |
| 5. | *If det (S)* $\neq 0$ *then* |
| 6. | *Break* |
| 7. | *End if* |
| 8. | *End while* |
| 9. | *Return S* |
| 10. | *End fun* |

Before the data is processed, the encryption of training data is done with a key that is certainly not shared. Subsequently, the model based on machine learning would have approached only ciphertext data (the encrypted form of data), whereas the plaintext data (actual data) are kept secretive on the data provider side separated from the processing unit. Lastly, the MORE encoding strategy with the homomorphic principles is applied (Table 4), which directly supports floating-point calculation. With the help of the uniform function of the MORE coding system, all the operations performed on the network are designed to ensure the application of the ciphertext data, and it can be directly trained on the ciphertext data. This creates a model that provides an encrypted prediction that only the owner of the secret key can decrypt. When the training phase is complete, you can use the coded model to predict a new cryptographic instance (the inference phase). Herewith the same key the input samples are encoded used in the training phase. MORE cryptographic systems rely on symmetric keys. Therefore, the decoding of the ciphertext data and the encoding of the plaintext data both are done with the secret key.

## VII. PROPOSED MODEL

There are many procedures that are used in ANN for this purpose. This research has explained a new method for predicting the gender through their voice with a feedforward propagation neural network over encryption data of plaintext into ciphertext. The proposed method is based on MORE

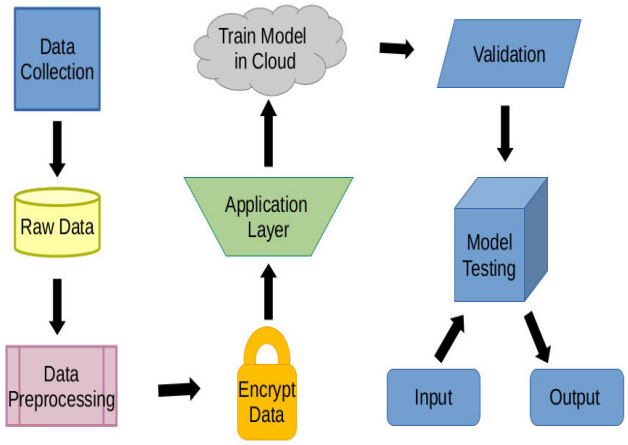



**FIGURE 3.** General framework.

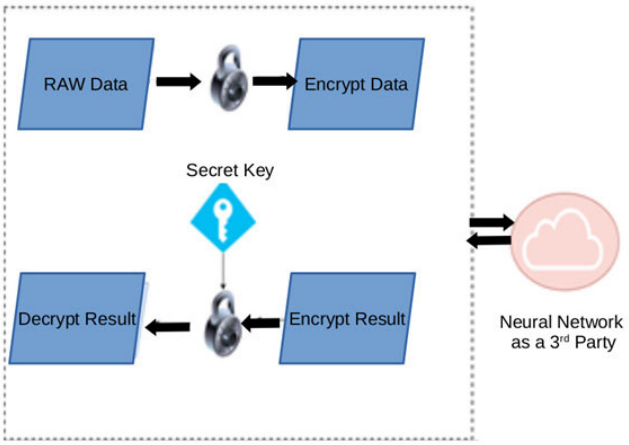



**FIGURE 4.** Workflow of proposed homomorphic encryption in neural network (NN).

homogeneous encryption technology and can directly train and develop a simple neural network model for homomorphically encrypted data. There are two phases first is the training phase and the second is validate phase, these two steps are linked one-to-one over a cloud environment. Fig. 3 shows the general framework of this research.

### A. METHOD

We can further develop the functionality of the NN network model by using the homomorphic properties of the MORE scheme to complete the work of coding data. The proposed workflow on homomorphic encryption and NN is shown in Fig. 4. In the processing stage, the trained data is encoded using a secret key that is not shared. Then, the neural network model can only access the encoded copy of the information or data (cipher encrypted text), and the actual data (plain text) is separated from the processing unit and kept secret from the data provider.

Finally, it directly supports floating-point calculation with the help of the homomorphic function of the MORE encrypting scheme, and all the processes achieved on the

| Attributes | Full Form |
|---|---|
| meanfreq | mean frequency (in kHz) |
| sd | standard deviation of frequency |
| Median | median frequency (in kHz) |
| Q25 | first quantile (in kHz) |
| Q75 | third quantile (in kHz) |
| IQR | interquantile range (in kHz) |
| skew | skewness |
| Kurt | kurtosis |
| sp.ent | spectral entropy |
| sfm | spectral flatness |
| mode | mode frequency |
| centroid | frequency centroid |
| peakf | peak frequency (frequency with highest energy) |
| meanfun | average of fundamental frequency measured across acoustic signal |
| minfun | minimum fundamental frequency measured across acoustic signal |
| maxfun | maximum fundamental frequency measured across acoustic signal |
| meandom | average of dominant frequency measured across acoustic signal |
| mindom | minimum of dominant frequency measured across acoustic signal |
| maxdom | maximum of dominant frequency measured across acoustic signal |
| dfrange | range of dominant frequency measured across acoustic signal |
| modindx | modulation index. |
| Label | male or female |

**FIGURE 5.** Attributes.

network are designed to confirm the application of the cipher-text data, and the network is able to train directly on the encrypted text information. This creates a model that gives an encoded prediction that merely the holder of the secret key can decrypt. When the training stage is complete, use the encoded model to predict new cipher instances. Here the input sample is encoded using the same key used in the training stage. MORE encryption schemes rely on symmetric keys to encrypt plain-text data and decrypt cipher-text data.

### B. DATASET

We took a dataset for voice recognition from kaggle.com (The dataset which is used to check the validity of the proposed methodology is available at link https://www.kaggle.com/primaryobjects/voicegender) it contains feature base data which is based on acoustic properties of speech and voice. The dataset includes 3168 recorded male and female voice samples, 21 attributes, and a few missing values that we calculate in preprocessing. The output variable has two classes that are identified as the voice of a man or woman. The attributes of our dataset are shown in Fig. 5.

### C. PRE-PROCESSING

Prior to primary processing, the sample data and group data are formulated which is required to build the set-up of the ANN model. To measure the noise of the signal before and after in given figures, encrypt data and store it in the cloud platform. For this sample data, we take built-in values for noise is noissin variable. After data collection, three basic issues are focused on for training using an artificial neural

network. The missing data problem is the first problem, and this data is interchanged by the ordinary immediate value. Second normalized data, and lastly randomized our data. A mean method used to calculate the missing values is formulated as:

$$T(c) = \begin{cases} \text{mean}(c), & \text{if } c = \text{null} \\ c, & \text{otherwise} \end{cases}$$

### D. APPLICATION LAYER

Next to pre-processing the application training layer is applied. It is further divided into two sub-layers, the forecast layer, and the enactment valuation layer. The forecast layer recycled the Adapted Feed-Forward neural network. In feed-forward propagation with the purpose of generating particular output, the data must not be fed in the back direction throughout the output generation, the input data should be flow only in the forward direction. Or else the output might not ever be produced as this will cause a rotation. The forecast layer consists of additional three layers, including the input layer, output layer, and hidden layer. Generally each layer is interrelated in a feed-forward manner. Each neuron of the heading layer has a direct association with the neurons of the successive layer. The input layer has contains 21 neurons; in the output layer 30 hidden neurons are involved that why there is simply one output.

Here, use sigmoid (x) function as an activation function, hidden layer s(x)= sigmoid (x) input inscribed as in Eq. (3).

$$\S_J = \sum_{i=1}^{m} (U_{ij} * \infty_i) + b_i \tag{3}$$

The hidden layer of the projected structure with the Sigmoid Function (SF) is presented in Eq. (4)

$$\partial_j = 1/(1 + e^{-\$j}) \tag{4}$$

where $k = 1, 2, 3, \ldots\ldots n$.

From the output layer input is occupied is presented in Eq. (5)

$$\$_k = b_2 + \sum_{j=1}^{n} (\beta_{jk} + \partial_j) \tag{5}$$

The Activation Function (AF) of output layer is presented in Eq. (6).

$$\partial_{k=} 1/(1 + e^{-\$k}) \tag{6}$$

where $k = 1, 2, 3, \ldots\ldots q$.

Fault in back propagation is show as in Eq. (7).

$$\infty - \frac{1}{2} \sum (\Upsilon - \partial_k)^2 \tag{7}$$

Afterward, calculate the enactment of the predicted layer in expressions of means square error and accuracy. If mandatory learning criteria are not attained, in that case, retrained the prediction layer. When learning standards are achieved, now move forward for validation purposes and the trained model is stored on the cloud.

Generally, training is usually implemented using back-propagation as a matrix. Back propagation is a process for the computing of all the essential data by means of the single forward pass at layers (by just calculating all the values of activation), and single backward pass (by calculating data in the network in a backwards direction) to compute all the necessary data in sequence. Back propagation is the center of neural network training. Appropriate training of the loads guarantees lesser rates of errors, creating the process consistent by enhancing its generality.

We tried to find the time-complexity of a training proposed method using a training sample and nodes i, j, k, l with n iterations. The result is o(nt * (iJ + jK + Kl)) o(nt * (iJ + jK + Kl)). We consider the most naive formula of matrix development with cubic time-complexity. It uses the adaptive Bayesian algorithm. The results of stochastic modifications and Bayesian adaptations should be the same. Also, with thrust optimization, the time-complexity of the algorithm is not affected because the additional matrix operations are all element-based and thus have the same time complexity.

### E. VALIDATION LAYER

When trained data is saved on the cloud, the validation step occurred which is further divided into two-layer, the prediction layer, and the data acquisition layer. The prediction layer contains the trained data and after that evaluation and calculation of received data occurred and forecasts the gender. The data acquisition layer contains input data, as specified earlier is similar.

## VIII. SIMULATION AND RESULTS

Machine learning procedure has been applied to the datasets and the MATLAB tool is used for simulations. In the machine learning method, the dataset contains 3168 samples, in training 70% of data is utilized (2217 samples) whereas for testing and validation remaining 30% of data is used (950 samples). Dataset is encrypted and has stored in the cloud when we want to decrypt data (ciphertext data) from the cloud we use HE. To evaluate the performance of our proposed model we train the above-mentioned datasets on different epochs and applied three different algorithms Levenberg-Marquardt (LM), Bayesian Regularization (BR), and Scale Conjugate Gradient (SCG) of ANN using the MATLAB tool. Regularization in order to enhance the generalization of network is utilized by the LM algorithm of ANN. Regularization includes variation in the function of performance. The sum of the square of the error in this is the performance function. In the SCG algorithm, along the conjugate gradient, a search is made in the gradient direction to conclude the size of steps along that line that reduces the performance function. The benefit of a BR ANN is its capability to show potentially complex interactions, which means that it will be able to use in computable work to deliver a robust model.

When we train our model, we identify that BR trained accurately rather than the other two algorithms of the neural network. The results of simulations in Figure 6 show that when we trained the dataset on the cloud by using neural network its state regression plot is built and training evolution results are obtained.

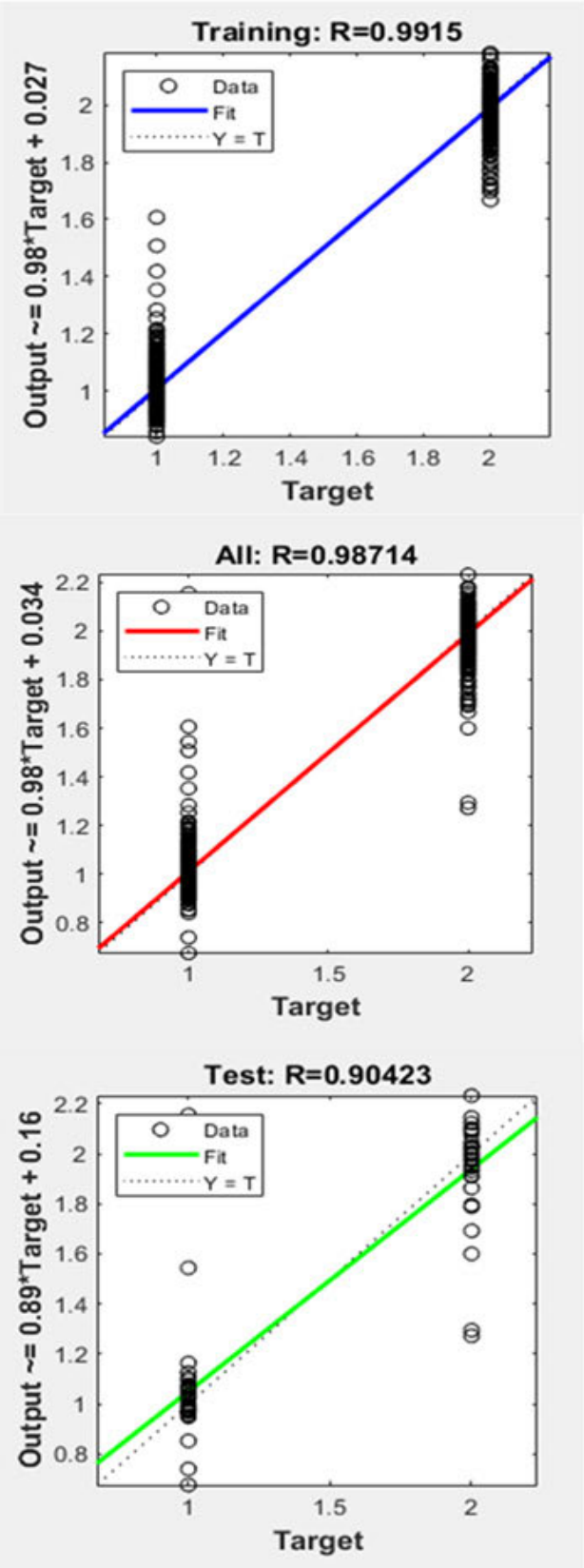



**FIGURE 6.** Training state regression plot and training evolution result of the dataset on cloud using neural network.

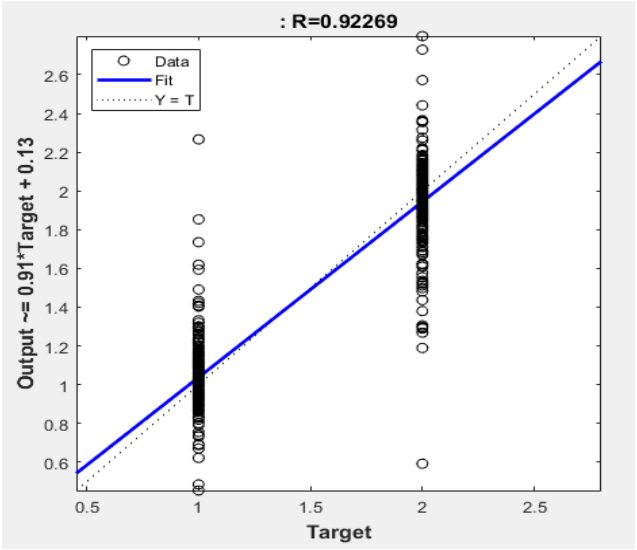



**FIGURE 7.** Validation result of the dataset on cloud using neural network.

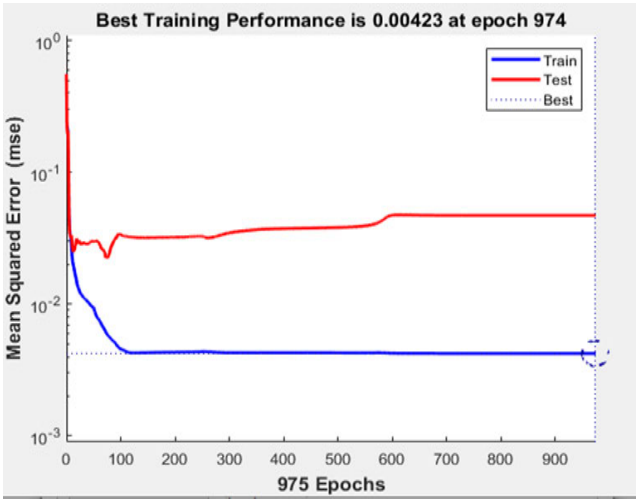
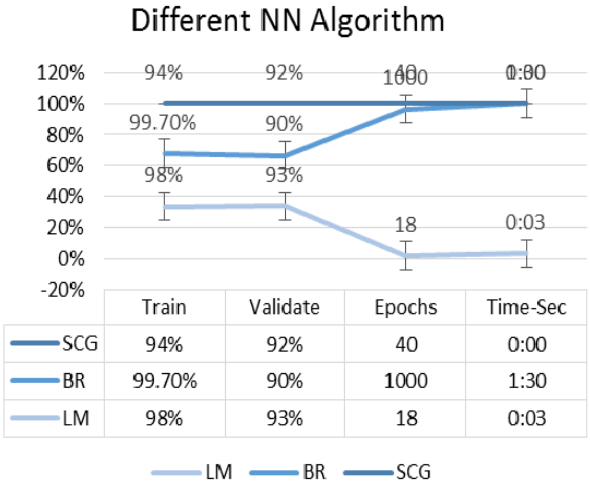



**FIGURE 8.** Performance evaluation of the dataset on cloud using neural network.

Figure 7 shows the validation/test result of the dataset on the cloud using a neural network algorithm. We applied this to validate the performance of the network. By training and testing datasets concerning targets, the regression plots show the outputs of the network. The data fall along a 45- degree line, a perfect fit, where the targets are equal to the outputs of the network. The fit is practically good for all data sets, with values of R in all cases is 0.9 or above for this problem, for more accurate results we can retrain the datasets. After retraining improved results may produce because the initial burden and preferences of the network will change.

Figure 8 shows the performance evaluation of the Bayesian Regularization algorithm. The validation result should be less than the training result, in this paper the results are accurate for the training and testing set. Training stops when a test error occurs in 6 iterations at 975 epochs as shown in the




| | Train | Validate | Epochs | Time-Sec |
|---|---|---|---|---|
| SCG | 94% | 92% | 40 | 0:00 |
| BR | 99.70% | 90% | 1000 | 1:30 |
| LM | 98% | 93% | 18 | 0:03 |

**FIGURE 9.** Comparison of neural network algorithms.

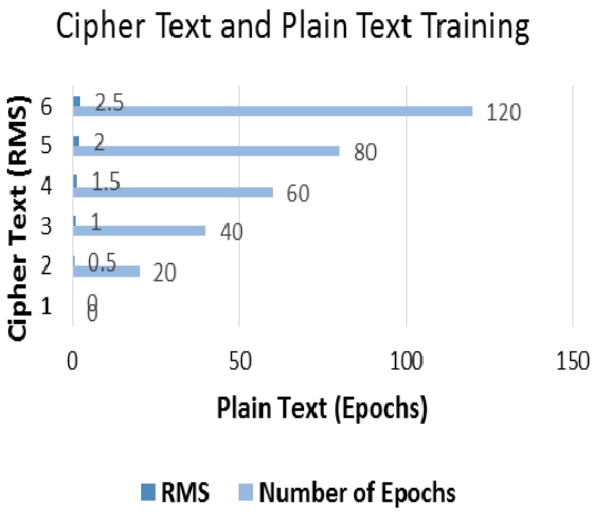



**FIGURE 10.** Development of the training for encoded networks on ciphertext data.

figure. In this simulation, the results are justified by the following considerations.

- The error of the finished isometric square is very small.
- Test setup errors and verification setup errors have similar characteristics.
- Redundancy 975 (which leads to the best validation performance) does not cause significant overfitting.

### A. PERFORMANCE

When we train the dataset using LM, BR, and SCG performance, training, and validation, results are given in Figure 9. The results show that as compared to SCG, LM gives very low performance and in contrast with BR, SCG performance is low.

To calculate the performance of the planned neural network model for privacy protection, we investigated two criteria: reliability and applicability in the speech place function library. Therefore, for each use case, we applied the model to both encrypted (encrypted text) and unencrypted (plain text) data to analyze the performance and results of the data-driven model. The results of the two cases are compared to analyze and measure the ability of the privacy model to maintain performance. In addition to dependability, an alternative factor that acts as a significant feature in defining the eligibility of confidentiality of a model to work in real-world processes is runtime. Consequently, the period was analyzed in detail and the conclusion and timing of the training were reported.

The results of the analysis show that the data security is guaranteed based on the homogeneous coding, and the NN data analysis can be achieved effectively. Research also shows that the data-driven model has been optimized in the same way in both encrypted and non-encrypted versions. To demonstrate the network capability to acquire the ciphertext data, after decoding the training error in the classification task is shown in Figure 10, where the number of epoch/iteration is shown on the x-axis and RMS shown in the y-axis. This graph represents the network's ability to learn from the ciphertext data and represents the loss of training due to regression processes after decoding.

Similarly, the accuracy of training of the ANN model that retains the personal information provided has also been developed. For getting more accurate results, we choose the Bayesian Regulation Algorithm in NN because Bayesian organization approaches can be cast off to answer prediction problems. This is because, according to the consequences of the investigation, the fault rate is very low and the results are already near to the preferred goal information. An evaluation of the three network structural design simulations used shows that the design model is the best because it takes very little time and results in mean square error testing and performance. This is superior to other models. Our data set is used for classification and we have two categories in our labels. This is why the other two algorithms do not give more reasonable results than Bayes' rule. The BR enhancement technology is superior to the other approaches. To do this, we used the BR algorithm to train our dataset at multiple intervals and then produce the best outcome for the security system in the cloud; as you can see that at 1000 epochs, it gives them a more relevant and accurate result. The results obtained after decoding are shown in Figure 11. Where the number of epoch/iteration is shown on the y-axis and time in sec is shown on the x-axis.

### B. CONCERNS REGARDING SECURITY

Even though the MORE encryption scheme has many benefits regarding directness, practicality and has some major features that go with machine learning privacy-preserving but in corresponding to other HE schemes its offers restricted security. The linear nature of MORE is providing the most important security concern because other common encryption schemes over large numbers are centered on strongly modular arithmetics and nonlinear functions. This linear nature of MORE may possibly allow someone to identify

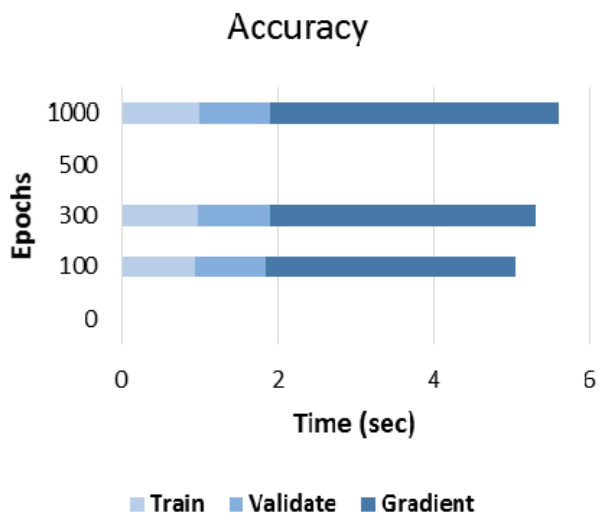



**FIGURE 11.** Development of the accuracy after training on cipher text data.

**TABLE 5.** Comparison with state of the art proposed techniques.

| No | Paper | Approaches | Dataset | Accuracy |
|---|---|---|---|---|
| 1. | Security and Privacy of Cloud- and IoT-Based Medical Image Diagnosis Using Fuzzy Convolutional Neural Network [39] | fuzzy convolutional neural network | Medical image dataset | 99% |
| 2. | Applying Deep Neural Networks over Homomorphic Encrypted Medical Data [40] | Convolutional Neural Network (CNN) | MNIST Dataset | 98.2% |
| 3. | Proposed Model | ANN | Voice Dataset | 99.70% |

the secret key by accessing a relatively large amount of encrypted and non-encrypted values in pairs. Otherwise specified, the secret key might be identified by an optimization problem if an adequately large amount of ciphertext-plaintext data pairs are present. For some specific privacy-preserving applications the MORE scheme remains a possible solution even though it is not as much secure as other HE schemes. Subsequently, MORE could be applied in settings where the key is never revealed, for example on the data provider side the raw data is kept private whereas to an external computing service ciphertext data are uploaded. The primary drawback of this arrangement is that it only permits encoding comparatively minor digits. More unambiguously, the encryption procedure requires an exponent, and the exponent is the message to be encoded. So even with a multi-precision computational library, this process will still overflow. It is found that using 1024-bit integers can only encode numbers up to $10^3$. This restriction develops even more significantly when accomplishing arithmetic operations on encrypted information. This means that it is impossible to manage if the encoded number is too large to perform a particular procedure. Performance presenting that the suggested scheme can resist the key related different kinds of attacks. Certainly, the planned methodology in contrast to the static technique employs the dynamic approach which is used by the current algorithms of symmetric homomorphic encryption.

Hence, the suggested technique against the weak keys delivers a good degree of resistance. Furthermore, the difference in the secret key after every pause and produces an altered set of dynamic key and therefore stop the unfortunate key disclosure.

### C. COMPARISON WITH PREVIOUS PROPOSED TECHNIQUES

Comparison with the previous state-of-the-art proposed techniques shows that our proposed model of MORE encrypting schemes into neural networks has shown improved accuracy. Table 5 shows the comparison in Security and Privacy of Cloud- and IoT-Based Medical Image Diagnosis Using Fuzzy Convolutional Neural Network techniques [39], Applying Deep Convolutional Neural Networks over Homomorphic Encrypted Medical Data techniques [40] and proposed technique.

## IX. CONCLUSION AND FUTURE WORK

In recent years, the demand for cryptographic techniques has increased that is suitable for data-driven models to address privacy-related issues. Homomorphic encoding scheme MORE that is Matrix-based was suggested in machine learning models for privacy-preserving estimation. HE technique is worked with private keys to encode and decode information, but it differs completely from other encryption techniques as it can preserve algebraic properties and perform various operations directly on the encoded data e.g. ciphertext data, without the need to access decoded data e.g. plain text information and encryption key.

We have shown the potential to incorporate MORE encrypting schemes into neural network models as results display improved runtime performance and accuracy. We focus on standard application in computer vision (speech frequency recognition) and classification of encrypted data on the basis of voice to assess the practicality of the network by operating directly on encoded data. We performed both training and validation steps and it shows both steps can be implemented on MORE data that is homomorphically encrypted. The results specify that the suggested encrypting scheme creates a fairly small computational burden and only slightly increases in runtimes; the important thing is that they allow direct operations on floating-point numbers that represent the main property of artificial neural networks.

Compared to the standard techniques, the MORE encoding technique provides less security, but it is one of the few schemes that can be used in real-world applications. In summary, it is shown that the HE method built on linear transformation has huge potential in simplifying the data distribution and allocating data to third parties for data analysis in regulatory domains, nonetheless, this is achieved at the expense

of poor security. Modifying the original HE scheme, being able to perform calculations directly on rational numbers (a prerequisite for machine learning models) puts security at risk. The proposed solution was promising at first, but for practicality, further improvements are required to increase the security of the system.

## REFERENCES

[1] S. L. Nita and M. I. Mihailescu, "On artificial neural network used in cloud computing security—A survey," in *Proc. 10th Int. Conf. Electron., Comput. Artif. Intell. (ECAI)*, Jun. 2018, pp. 1–6.
[2] Z. Zhang *et al.*, "Artificial intelligence in cyber security: Research advances, challenges, and opportunities," *Artif. Intell. Rev.*, Mar. 2021, doi: 10.1007/s10462-021-09976-0.
[3] A. Agarwal, M. Khari, and R. Singh, "Detection of DDOS attack using deep learning model in cloud storage application," *Wireless Pers. Commun.*, Mar. 2021, doi: 10.1007/s11277-021-08271-z.
[4] M. Abadi, A. Chu, I. Goodfellow, H. B. Mcmahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 308–318.
[5] M. Choraś and M. Pawlicki, "Intrusion detection approach based onoptimised artificial neural network," *Neurocomputing*, vol. 452, pp. 705–715, Sep. 2021.
[6] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 1–35, 2018.
[7] B. Li and D. Micciancio, "On the security of homomorphic encryption on approximate numbers," in *Advances in Cryptology—EUROCRYPT 2021* (Lecture Notes in Computer Science), vol. 12696, A. Canteaut and F. X. Standaert, Eds. Cham, Switzerland: Springer, Oct. 2021, doi: 10.1007/978-3-030-77870-5_23.
[8] F. Boemer, A. Costache, R. Cammarota, and C. Wierzynski, "NGraph-HE2: A high-throughput framework for neural network inference on encrypted data," in *Proc. 7th ACM Workshop Encrypted Comput. Appl. Homomorphic Cryptogr. (WAHC)*, 2019, pp. 45–56.
[9] P. Mishra, R. Lehmkuhl, A. Srinivasan, W. Zheng, and R. A. Popa, "Delphi: A cryptographic inference service for neural networks," in *Proc. 29th USENIX Security Symp. (USENIX Secur.)*, 2020, pp. 2505–2522.
[10] M. S. Riazi, C. Weinert, O. Tkachenko, E. M. Songhori, T. Schneider, and F. Koushanfar, "Chameleon: A hybrid secure computation framework for machine learning applications," in *Proc. Asia Conf. Comput. Commun. Secur.*, May 2018, pp. 707–721.
[11] Q. He and H. He, "A novel method to enhance sustainable systems security in cloud computing based on the combination of encryption and data mining," *Sustainability*, vol. 13, no. 1, p. 101, Dec. 2020.
[12] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 201–210.
[13] A. Dalskov, D. Escudero, and M. Keller, "Secure evaluation of quantized neural networks," 2019, *arXiv:1910.12435*. [Online]. Available: http://arxiv.org/abs/1910.12435
[14] V. M. Lidkea, R. Muresan, and A. Al-Dweik, "Convolutional neural network framework for encrypted image classification in cloud-based ITS," *IEEE Open J. Intell. Transp. Syst.*, vol. 1, pp. 35–50, 2020.
[15] A. H. Shaikh and B. Meshram, "Security issues in cloud computing," in *Intelligent Computing and Networking* (Lecture Notes in Networks and Systems), vol. 146, V. E. Balas, V. B. Semwal, A. Khandare, and M. Patil, Eds. Singapore: Springer, 2021, doi: 10.1007/978-981-15-7421-4_6.
[16] T. Agrawal and S. Singh, "Analysis of security algorithms in cloud computing," in *Proc. 3rd Int. Conf. Comput. Sustain. Global Develop. (INDIACom)*, Mar. 2016, pp. 106–108.
[17] A. J. Ferrer, D. G. Pérez, and R. S. González, "Multi-cloud platform-as-a-service model, functionalities and approaches," *Proc. Comput. Sci.*, vol. 97, pp. 63–72, Jan. 2016.
[18] C. Gentry and S. Halevi, "Implementing gentry's fully-homomorphic encryption scheme," in *Advances in Cryptology—EUROCRYPT 2011* (Lecture Notes in Computer Science), vol. 6632, K. G. Paterson, Ed. Berlin, Germany: Springer, May 2011, doi: 10.1007/978-3-642-20465-4_9.
[19] E.-Y. Ahmed and M. D. Elkettani, "Fully homomorphic encryption: State of art and comparison," *Int. J. Comput. Sci. Inf. Secur.*, vol. 14, no. 4, p. 159, 2016.
[20] S. S. Sathya, P. Vepakomma, R. Raskar, R. Ramachandra, and S. Bhattacharya, "A review of homomorphic encryption libraries for secure computation," 2018, *arXiv:1812.02428*. [Online]. Available: http://arxiv.org/abs/1812.02428
[21] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Advances in Cryptology—ASIACRYPT 2017* (Lecture Notes in Computer Science), vol. 10624, T. Takagi and T. Peyrin, Eds. Cham, Switzerland: Springer, 2017, doi: 10.1007/978-3-319-70694-8_15.
[22] J. Fan and F. Vercauteren, "Somewhat practical fully homomorphic encryption," *IACR Cryptol. ePrint Arch.*, vol. 2012, p. 144, Mar. 2012.
[23] S. Halevi and V. Shoup, "Bootstrapping for HElib," *J. Cryptol.*, vol. 34, no. 1, pp. 1–44, Jan. 2021.
[24] J. Mancuso, "Privacy-preserving machine learning 2018: A year in review," 2018. [Online]. Available: https://medium.com/dropoutlabs/privacy-preserving-machine-learning-2018-a-year-in-review-b6345a95ae0f
[25] P. V. Parmar, S. B. Padhar, S. N. Patel, N. I. Bhatt, and R. H. Jhaveri, "Survey of various homomorphic encryption algorithms and schemes," *Int. J. Comput. Appl.*, vol. 91, no. 8, pp. 26–32, Apr. 2014.
[26] H. Chung and M. Kim, "Encoding rational numbers for fhe-based applications," IACR Cryptol. ePrint Arch., Tech. Rep. 2016/344, Mar. 2016.
[27] P. Mohassel and Y. Zhang, "SecureML: A system for scalable privacy-preserving machine learning," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2017, pp. 19–38.
[28] A. Alabdulatif, I. Khalil, and X. Yi, "Towards secure big data analytic for cloud-enabled applications with fully homomorphic encryption," *J. Parallel Distrib. Comput.*, vol. 137, pp. 192–204, Mar. 2020.
[29] J. Park, D. S. Kim, and H. Lim, "Privacy-preserving reinforcement learning using homomorphic encryption in cloud computing infrastructures," *IEEE Access*, vol. 8, pp. 203564–203579, 2020.
[30] H. Pang and B. Wang, "Privacy-preserving association rule mining using homomorphic encryption in a multikey environment," *IEEE Syst. J.*, vol. 15, no. 2, pp. 3131–3141, Jun. 2021.
[31] Li, X. Kuang, S. Lin, X. Ma, and Y. Tang, "Privacy preservationfor machine learning training and classification based on homomorphicencryption schemes," *Inf. Sci.*, vol. 526, pp. 166–179, Jul. 2020.
[32] D. Vizár and S. Vaudenay, "Cryptanalysis of chosen symmetric homomorphic schemes," *Studia Sci. Math. Hungarica*, vol. 52, no. 2, pp. 288–306, 2015.
[33] B. Tsaban and N. Lifshitz, "Cryptanalysis of the more symmetric key fully Homomorphic encryption scheme," *J. Math. Cryptol.*, vol. 9, no. 2, pp. 75–78, 2015.
[34] K. Hariss, H. Noura, and A. E. Samhat, "Fully enhanced homomorphic encryption algorithm of MORE approach for real world applications," *J. Inf. Secur. Appl.*, vol. 34, pp. 233–242, Jun. 2017.
[35] H. Chabanne, A. de Wargny, J. Milgram, C. Morel, and E. Prouff, "Privacypreserving classification on deep neural network," *IACR Cryptol. ePrint Arch.*, vol. 2017, p. 35, Mar. 2017.
[36] E. Hesamifard, H. Takabi, and M. Ghasemi, "CryptoDL: Deep neural networks over encrypted data," 2017, *arXiv:1711.05189*. [Online]. Available: http://arxiv.org/abs/1711.05189
[37] L. Xiao, O. Bastani, and I. L. Yen, "An efficient homomorphic encryption protocol for multi-user systems," *IACR Cryptol. EPrint Arch.*, vol. 2012, p. 193, Apr. 2012. [Online]. Available: http://eprint.iacr.org/2012/193
[38] A. Kipnis and E. Hibshoosh, "Efficient methods for practical fully homomorphic symmetric-key encrypton, randomization and verification," *IACR Cryptol. ePrint Arch.*, vol. 2012, p. 637, Nov. 2012.
[39] J. Deepika, C. Rajan, and T. Senthil, "Security and privacy of cloud- and IoT-based medical image diagnosis using fuzzy convolutional neural network," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–17, Mar. 2021.
[40] A. Vizitiu, C. I. Niţă, A. Puiu, C. Suciu, and L. M. Itu, "Applying deep neural networks over homomorphic encrypted medical data," *Comput. Math. Methods Med.*, vol. 2020, pp. 1–26, Apr. 2020.

**MUHAMMAD USMAN SANA** received the B.E. degree in information technology from the University of Engineering and Technology, Taxila, Pakistan, in 2006, and the M.S. degree in communication engineering from the Chalmers University of Technology, Sweden, in 2010. He is currently pursuing the Ph.D. degree with the Xi'an University of Science and Technology, China. His current research interests include cloud computing, networks security, and safety mechanism.

**ZHANLI LI** received the B.S. degree from Xianyang Normal University, Xi'an, China, in 1984, the M.S. degree from the Xi'an University of Science and Technology, Xi'an, in 1989, and the Ph.D. degree from Xi'an Jiaotong University, Xi'an, in 1997. He is currently a Professor and the Dean of the School of Computer, Xi'an University of Science and Technology, and a Professor of engineering at Iwate University, Japan. He has authored more than 50 technical papers for conferences and journals, and holds more than ten invention patents. His research interests include artificial intelligence, danger perception, and image identification.

**FAWAD JAVAID** was born in Sialkot, Pakistan. He received the B.S. degree in electronic engineering from International Islamic University, Pakistan, in 2012, and the M.S. degree in electrical engineering from the University of Gujrat, Pakistan, in 2016. He is currently pursuing the Ph.D. degree with the Xi'an University of Science and Technology, China. He was a valuable member of teaching faculty in multiple leading universities of Pakistan, from 2012 to 2018. His research interest includes the domain of wireless communication and networks. He is also an Affiliated Member of the Pakistan Engineering Council.

**HANNAN BIN LIAQAT** received the B.S. degree in information technology and the M.S. degree in computer networks from the COMSATS Institute of Information Technology, Lahore, Pakistan, in 2006 and 2009, respectively, and the Ph.D. degree from the Dalian University of Technology, Dalian, China, in 2016. From 2009 to 2011, he was a Lecturer with the Department of Information Technology, University of Gujrat, Gujrat, Pakistan. Since 2016, he has been working with the Department of Information Technology, University of Gujrat, as an Assistant Professor and a Ph.D. and M.Phil. Supervisor. He has a number of publications to his credits in international journals and conferences. His research interests include *ad hoc* social networks, the IoT, cloud computing, mobile computing, and social computing. Furthermore, he also worked as the technical program chair in number of conferences. He is a reviewer of several international journals and conferences.

**MUHAMMAD USMAN ALI** received the M.S. degree in computer networks engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2008, and the Ph.D. degree from Yeungnam University, South Korea, in 2018. He is currently working with the Department of Computer Science, University of Gujrat, Pakistan. His current research interests include multi-sensor fusion-based indoor positioning systems, Wi-Fi fingerprinting, indoor navigation and mapping, computer vision, and computer networks technologies.