International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Adaptive Serial Cascaded Deep Network-based Data Deduplication Mechanism with Hyper-Elliptic Curve Cryptography for Encryption in Cloud Environment

Mr. Manjunath Singh H.*[1], Dr. Tanuja R.[2]

Submitted: 03/02/2024 **Revised:** 11/03/2024 **Accepted:** 17/03/2024

Abstract*:* Cloud storage services utilize deduplication to optimize capacity and minimize bandwidth demands. This process efficiently reduces redundant data to a single instance, thereby conserving storage space. Deduplication is particularly effective when multiple users upload identical information to the cloud. However, deduplication poses challenges related to security and copyright issues. Implementing secure deduplication can significantly cut down on both storage and communication costs in cloud services, making it highly relevant in the era of big data. Systems that verify the proof-of-ownership allow individuals who have uploaded the same data to credibly assert their ownership to the cloud service. However, the common practice of encrypting data before uploading it for privacy reasons complicates deduplication efforts because encryption introduces randomness that prevents identifying duplicates. To overcome this, various schemes have been introduced that permit users to encrypt data with a common key for identical data sets. Nevertheless, many of these schemes are susceptible to security flaws, particularly not addressing the frequent changes in data ownership in a dynamic cloud storage environment. Therefore, creating a secure data deduplication model that overcomes the limitations of current approaches is essential. The implemented framework consists of data collection, deduplication phase and encryption. Initially, attributes likes “filename, size, block name, size, file-type hash tag, file location, file updated date and data pattern” are used for the deduplication process. Next, the collected data is provided as the input to the Optimized Serial Cascaded Deep Network (OSCDN)-based data deduplication model, which is the fusion of “Deep Belief Network (DBN) with Dilated Convolution Long Short Term Memory (DConv-LSTM)”, Here the parameters of OSCDN is tuned using “Enhanced Red-Tailed Hawk algorithm (ERTH)”. Further, the de-duplicated data is encrypted using “Hyper-Elliptic Curve Cryptography with Optimal Key (HECC-OK)”. In this setup, the ERTH algorithm selects keys in the most optimal manner. Subsequently, the encrypted data is stored on the cloud platform. The developed architecture then undergoes several experimental validations to showcase its enhanced performance rate relative to traditional deduplication methods.

Keywords: *Data Deduplication; Cloud Environment; Optimized Serial Cascaded Deep Network; Enhanced Red-Tailed Hawk algorithm; Hyper-Elliptic Curve Cryptography with Optimal Key.*

1. Introduction

The Internet of Things (IoT) is a burgeoning concept with significant applications in the manufacturing sector. In these environments, “Wireless Sensor Networks (WSN) and Wireless Sensor-Actuator Networks (WSAN)” are employed to gather data on various aspects, such as "energy savings, air quality management, predictive maintenance, resource forecasting, and product planning" [9]. In the context of Smart Factories utilizing IoT, optimizing energy consumption and production time are key priorities [10]. A considerable portion of the data outsourced in IoT systems is redundant, leading to unnecessary storage costs [11]. The effective strategy to mitigate these costs is by purging duplicate data prior to its storage on cloud platforms. However, given that cloud servers are often owned by third parties, there's a valid concern regarding the security of sensitive information during the deduplication process [12]. Integrating data compression techniques within cloud storage solutions offers a way to significantly reduce storage demands and, by extension, lower the costs associated with data storage. Specifically, data deduplication is a method employed to detect and eliminate duplicate data within a storage system.

Deduplication streamlines storage by replacing duplicate data segments (either entire files or parts of files) with references to a single instance of that data already stored on the disk [14]. Unlike traditional compression techniques, deduplication can eliminate redundancies not just within a single file, but also across multiple files [15]. However, spotting these duplicates requires analyzing vast volumes of data, a process that is both computationally intensive and heavy on input/output (I/O) operations, potentially slowing down server performance significantly [16]. To alleviate the negative impact of deduplication on server performance, an effective strategy involves distributing the deduplication workload across multiple nodes within a storage cluster. By leveraging the collective computational power and storage capacity of these nodes, the challenges of data deduplication can be more manageably addressed [17]. One of the key technological challenges in this distributed approach is to achieve scalable

[1] *Associate Professor, Computer Science and Engineering*
[2] *Associate Professor, Computer Science and Engineering. tanujar.uvce@gmail.com*
** Corresponding Author Email: mansh24.singh@gmail.com*

performance that still maintains a system-wide data reduction ratio on par with centralized deduplication systems [18]. The optimal data deduplication ratio requires a global assessment and comparison of data, which becomes increasingly feasible as vast amounts of information are now stored in cloud environments [19].The inability of a data disk to identify and eliminate duplicate information can lead to unnecessary consumption of disk storage space. This redundancy not only occupies precious space but also impacts disk performance, affecting speed and other key performance metrics [20]. By strategically managing deduplication across distributed systems, it's possible to minimize these drawbacks and optimize storage efficiency.

Cloud platforms offer their users significant computational and storage benefits, enhancing efficiency and reducing operational costs. Integrating data compression techniques within these cloud services can significantly diminish the need for extensive storage space, further cutting down on data storage expenses [21]. However, managing large datasets can increase memory usage, raise costs, and extend CPU processing times for clients [22]. Data deduplication emerges as a solution to these challenges by efficiently identifying and retaining only essential data, thereby eliminating redundant data and maintaining high levels of security and privacy [23]. A critical limitation of this method is the ease with which both internal and external adversaries could potentially deduce the nature of the stored data [24] [25]. To overcome these challenges, an advanced and secure online data replication model for cloud storage has been developed.

The goals of the recommended model are detailed below:

- To design a deep learning-based data depuplication model in cloud environment that aims to efficiently identify and remove duplicate data within a storage system to optimize storage usage and reduce redundancy.
- To design ERTH for optimally selecting the parameters, this algorithm is inspired by the hunting behavior of RTH. This model is designed to find the most efficient keys that enhance security and minimize computational overhead.
- To create an OSCDN for data deduplication. This innovative model combines DBN with DConv-LSTM allows for efficient identification and removal of duplicate data. The parameters of OSCDN are optimally tuned using ERTH algorithm.
- To design an HECC-OK model for security privacy, this is known for providing high levels of security with optimal keys, making it efficient for cloud environments and reduces the time and memory. The selection of the optimal keys is done with the help of ERTH algorithm.
- To compare the performance data of the developed framework against the benchmarked traditional mechanisms. Use statistical methods to determine the significance of the observed differences.

The subsequent sections delve deeply into every facet of the proposed deep learning framework aimed at data deduplication and and encryption in cloud environment enhancement. Section II presents a review of existing literature in the realms of information deduplication and encryption techniques. Section III discusses the essential prerequisites for data deduplication within a cloud environment, including the specifics of input features for the proposed approach. Section IV offers a comprehensive examination of addressing data duplication in the cloud through an innovative adaptive serial cascaded deep network. Section V shifts the focus to bolstering data security through an advanced heuristic algorithm-enhanced HECC strategy. Sections VI and VII further explore the manipulation of numerical data and outline the deep learning-based strategy implemented for efficient data deduplication.

2. Literature survey

2.1. Related Works

In 2015, Luo *et al.* [1] have developed a state-of-the-art online storage solution was ingeniously crafted with distributed compression technology at its core. This platform stood out for its exceptional efficiency and scalability, achieved through a novel parallel data deduplication strategy designed to preserve duplication ratios effectively. At the heart of Boafft's operation was a refined data transmission system, which smartly exploited data similarities to enhance network efficiency and swiftly pinpointed the optimal storage destinations. Moreover, Boafft significantly boosted its performance by deploying in-memory similarity indexes on each server, thereby eliminating extensive random disk access and significantly speeding up the deduplication effort.

In 2016, Yan *et al.* [2] have developed system leveraging “Attribute-Based Encryption (ABE)” to address the challenge of managing encrypted data duplication in cloud storage while ensuring secure data handling. The performance and effectiveness of this model were rigorously evaluated through both analytical and practical means. Findings indicated that this approach was not only viable but also efficient and scalable, making it suitable for real-world applications. Employing deduplication techniques for encrypted data emerged as a critical strategy in delivering a secure, reliable, and efficient cloud storage service, especially vital for handling large-scale data operations.

In 2016, Mao *et al.* [3] have proposed “Performance-Oriented I/O Deduplication (POD)” replaced Space-Oriented I/O Deduplication (iDedup) as the preferred method for enhancing the I/O efficiency of primary storage devices within the Cloud, albeit at the expense of reduced capacity savings. To boost the storage system's efficiency and reduce the costs associated with deduplication, POD implemented a dual-strategy approach. This approach encompassed a flexible memory management scheme known as iCache, designed to alleviate recall contention amid bursty read and write traffic, and a request-based selective deduplication technique called Select-Dedupe, which aimed to lessen data fragmentation.

In 2020, Sharma *et al.* [4] have developed task allocation and secure data compression were efficiently executed across four distinct layers within a Fog-assisted Cluster-based Industrial IoT (IIoT) framework. The initial layer, known as the IoT device layer, was dedicated to data collection and bolstering security measures. In this layer, devices were securely connected to cloud services through the use of Elliptic Curve Cryptography-based Hybrid Amplifiers. For task clustering, a multi-objective Whale Optimization Algorithm (WOA) was deployed. To ensure data security during compression, the SHA-3 encryption standard was implemented within the fog layer. Prior to transmission, data was encrypted using an ECC-based Hybrid Model (HM) secret key. For efficient data retrieval and integrity verification within the cloud, a Merkle Hash Tree (MHT) was utilized.

In 2023, Muthunagai and Anitha [5] have presents the CTS-IIoT framework, a MHT plays a crucial role in managing time-series data through index-based deduplication of IIoT data stored in the

cloud. Ultimately, the proposed system leverages the Modified Distribution (MODI) method to pinpoint the most cost-effective location for accessing all storage resources within the cloud domain.

In 2021, Olufemi *et al.* [6] have proposed a faster server-aided approach for eliminating data duplication, incorporating an efficient verified key agreement. This technique ensured data singularity, privacy and consistency in IIoT environments and enabled secure data searches over encrypted content on a semi-trusted cloud platform.

In 2022, Muthunagai and Anitha [7] have proposed the "Adaptive Multi-Pattern Boyer Moore Horspool (AM-BMH) algorithm" and MT to extract insights from time series data. A significant challenge encountered was the high data transfer costs within the globally decentralized cloud system, which complicated data placement strategies. To overcome this, an innovative data insertion approach featuring optimized distribution was introduced.

In 2022, Vignesh and Preethi [8] have developed a novel method for eliminating duplicate data on the internet, aiding in the conservation of both bandwidth and storage space. Trial results demonstrated that the proposed method enhanced data security stored in the cloud while simultaneously addressing the primary deficiencies of existing systems. A solution was developed for both single-server memory and distributed storage systems, ensuring the privacy of information while also conserving space. The chunk information consistently generated encryption keys, leading to identical chunks being encrypted with the same ciphertext.

2.2. Problem statement

The cloud storage system provides the more storage space to preserve the private information of the user. It is easy to access the file based on the storage location. But, it is vulnerable to brute-force assault, which determines the plain text in the storage space corresponded to an attained cipher-text. Various researches were performed for data deduplication and the features and challenges of the data deduplication in discussed in Table. I. The research gap of the traditional data deduplication approaches is given below.

- Existing techniques employs single deep learning approaches, which causes integrity issues on the network. It can be resolved by utilizing an advanced deep learning network to enhance the generalization of the network in deduplication.
- In several traditional approaches it does not uses encryption approaches that leads to information misuse an affects the security of the network. This can be resolved by performing encryption approaches.
- Traditional models haven employed single technique for data deduplication it minimize the performance of the network and also it is hard to maintain the generated data. It can be resolved by utilizing cascaded deep learning technique.
- Several conventional models utilize encryption to generate the private key, but in certain cases it is difficult to maintain the network privacy. This issue can be resolved by utilizing a tuning approach to tune the generated key.

Table I: Features And Challenges Of The Existing Data Deduplication In Cloud Environment Approaches

Author [citation]	Methodology	Features	Challenges
Luo *et al.* [1]	Boafft	• This approach minimizes the overhead of system bandwidth. • This technique rapidly evaluates the storage location of the information.	• This technique is composite to organize and preserve.
Yan *et al.* [2]	ABE	• This model is highly secure for preserving the information. • This approach offers fine-grained access control for the framework.	• This technique lack straight forward attribute and in effective in the real world
Mao et al. [3]	POD	• This technique achieves improved efficiency is information preservation. • This approach considerably built the appearance and helps to preserving ability of the principal preservation framework in the cloud storage.	• This technique consumes high memory space for processing the information. • The effectiveness of the network does not relies on the energy competence and capacity.
Sharma *et al.* [4]	WOA	• This strategy helps to resolve various constrained or unconstrained tuning issues.	• This approach gets stucked on the local optimum. • It posses low convergence speed and reduced precision rate.
Muthunagai and Anitha[5]	MHT	• It assists to evaluate the integrity of information or communication. • This technique offers an advanced and easy way to organize the information.	• This technique faces difficulties in data addition and deletion. • It requires more memory space and processing time.
Olufemi *et al.* [6]	Data de-duplication scheme	• It allows sheltered searching of information next to the cipher-text on the cloud storage.	• This approach requires high computational cost • They are capable to identify only the deduplicate information stored on the cloud.

Muthunagai and Anitha [7]	BMH algorithm, and Merkle tree	• This approach offers improved performance as the length of pattern rises. • This technique utilizes less cloud storage space.	• The elimination of good-suffix sometime affects the network performance. • This requires more resources for processing.
Vignesh, R, and J. Preethi [8]	AES and RSA algorithms	• This technique utilize longer key size for data encryption • It is safe and dependable in transmitting the private data.	• This approach is susceptible to side-channel attacks. • It has low convergence speed.

3. Basic Requirement of Data Deduplication in Cloud Environment and Input Attribute Details for Proposed Work

3.1. Need for Data Deduplication in Cloud

Data deduplication is a powerful strategy for enhancing storage efficiency and reducing the overall volume of stored data. By identifying and eliminating redundant copies of data and replacing them with a single original and logical links for any subsequent references, deduplication significantly decreases the need for storage [8]. This process benefits from the use of hash values, unique identifiers that help identify redundant data chunks. The benefits of optimization extend beyond mere reduction in storage needs; include decreased network bandwidth requirements, lower costs and reduced energy consumption for storage systems. The flexibility of data deduplication makes it suitable for various stages of data storage and transmission, especially in cloud storage solutions. Cloud services leverage deduplication to streamline disaster recovery efforts, ensuring that only unique data is replicated post-deduplication. This approach not only accelerates replication times but also conserves network bandwidth. Additionally, deduplication proves crucial in backup and archival systems within cloud storage, reducing the physical storage footprint and minimizing network traffic. However, when considering the implementation of deduplication for primary storage, especially for dynamic data like computer images, it is vital to weigh the space-saving benefits against potential impacts on performance.

The ever-increasing volume of data generated and stored online necessitates efficient management strategies to optimize storage, reduce costs and ensure rapid data access and transfer. In this context, data deduplication emerges as a critical technology, particularly for cloud environments. Cloud-based services and storage systems face the dual challenge of scaling to meet growing demand while maintaining high levels of efficiency and performance [5]. This introductory exploration highlights the crucial need for data deduplication in cloud environments, underscoring its role in enhancing storage optimization, minimizing resource consumption and improving overall system performance and reliability.

3.2. Proposed System and its Description

Data deduplication, a compression technique, removes redundant copies of data, retaining only a single unique instance. By storing only one copy of the data, this method drastically decreases storage needs while still allowing access to the information when required. It is applicable across various storage environments, including networked storage, cloud storage and backup systems. The key advantage of data deduplication is the dramatic reduction in storage space required, potentially leading to cost savings and prolonging the lifespan of existing storage infrastructure. However, the process of identifying and removing duplicate data can incur computational costs, potentially impacting system performance, especially if deduplication occurs in real-time. As data volumes expand, the deduplication system must scale accordingly, maintaining efficiency and effectiveness in the face of growing storage needs. To address these issues, implementing the suggested model can improve the system's overall efficiency. Fig. 1 illustrates the proposed data deduplication approach in a cloud environment.

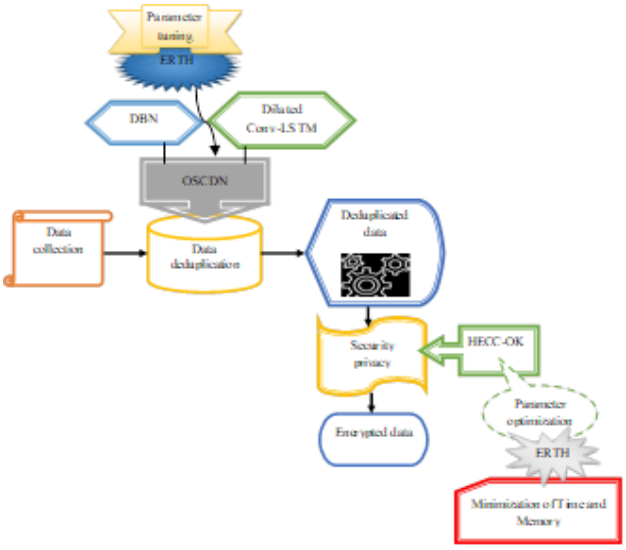

Fig. 1. Structural view of recommended data deduplication in cloud environment

The burgeoning data generated in cloud environments necessitates advanced strategies for storage optimization and redundancy reduction. To addressing this challenge, a sophisticated deep learning-based data deduplication model tailored for cloud settings is proposed. Data duplication, also known as data augmentation, is a technique used in deep learning and machine learning to artificially increase the size of a dataset by creating additional variations of the existing data. This process is particularly useful when the available dataset is small, as it helps improve the generalization and robustness of the machine learning model. This model aims to enhance storage efficiency by accurately identifying and eliminating duplicate data, thereby optimizing storage utilization and minimizing redundancy. Additionally, the ERTH algorithm is designed to optimally select parameters that bolster security while simultaneously reducing computational overhead. To advance data deduplication efforts, the OSCDN model is proposed. This cutting-edge model integrates DBN with DConv-LSTM networks, facilitating the effective identification and removal of duplicate data. The OSCDN model's parameters are

finely tuned using the ERTH algorithm to ensure optimal performance. In the realm of security privacy, the HECC-OK model is designed to offer robust security solutions. Known for its high-security levels with optimally selected keys, HECC-OK is particularly efficient for environments constrained by resources, significantly reducing time and memory requirements. The selection of these optimal keys is tuned through the ERTH algorithm, ensuring the security privacy model's effectiveness and efficiency.Finally validate the efficacy and efficiency of the developed framework, a comprehensive comparison against traditional data deduplication and encryption mechanisms is planned. This evaluation will employ statistical methods to analyze performance data, aiming to establish the statistical significance of the observed improvements.

4. Data Deduplication in Cloud done by Adaptive Serial Cascaded Deep Network

4.1. Deep Belief Network

This method [33] is characterized by a generative visual technique, utilizing multi-tiered neural network architecture to master the representations derived from the training dataset. Typically, this involves multiple hidden layers, an input layer, and an output layer positioned at the pinnacle, which delivers the ultimate data. The primary aim of this configuration is to enable the network model to reconstruct the input data using the features it has generated, by adjusting the weights of the nodes across different layers.

Eq. (1) delineates the combined distribution of the hidden and input layers within the DBN approach.

$$I\left(DF_z^{AE}, a^1, \cdots, a^l\right) = I\left(DF_z^{AE} / a^1\right)\left(\coprod_m^l DF\left(a^m / a^{m+1}\right)\right) \quad (1)$$

Here, the variable $D\left(a^m / a^{m+1}\right)$ dictates the conditional allocation between adjacent $m+1$ and m layers, where a^m represents the data vector of the m layer, l denotes the total number of hidden layers and DF_z^{AE} signifies the input information vector.

In the process, each subsequent layer pair $D\left(a^m / a^{m+1}\right)$ in the DBN model is trained using a RBM. This configuration treats the initial layer as a visible layer and the following layer as a hidden layer. Moreover, the RBM is structured as a bipartite, undirected graph, comprising two distinct layers. The determination of this structure is further elaborated in Eq. (2) and Eq.(3).

$$D\left(a^m / a^{m+1}\right) = \coprod_{k=1}^{n_m} I\left(a_k^m / a^{m+1}\right) \quad (2)$$

$$I\left(a^m = 1 / ja\right) = S\left(RT_k^m + \sum_{aa=1}^{n_{m+1}} mn_{aak}^m jk_{aa}^{m+1}\right) \quad (3)$$

Moreover, the model developed by the DBN is capable of autonomously learning the JK^m and RT_k^m matrices throughout the iterative process. This learning has been facilitated through the use of gradient descent, as detailed in Eq. (4) and (5).

$$JK_{ko}(s+1) = JK_{ko}(s) + \omega \frac{\partial \log\left(I(mm/a)\right)}{\partial wg_{ko}} \quad (4)$$

$$RT_{ko}(s+1) = RT_{ko}(s) + \omega \frac{\partial \log\left(I(mm/a)\right)}{\partial wg_{ko}} \quad (5)$$

Additionally, in the initial phase of the network model, all matrix values were initialized between the two stages of RBM for training. A bias value was also established to fine-tune the parameters according to the specified criteria. Consequently, the DBN architecture exhibits the capability to establish more intricate networking relationships. Fig. 2 shows the architecture view of DBN.

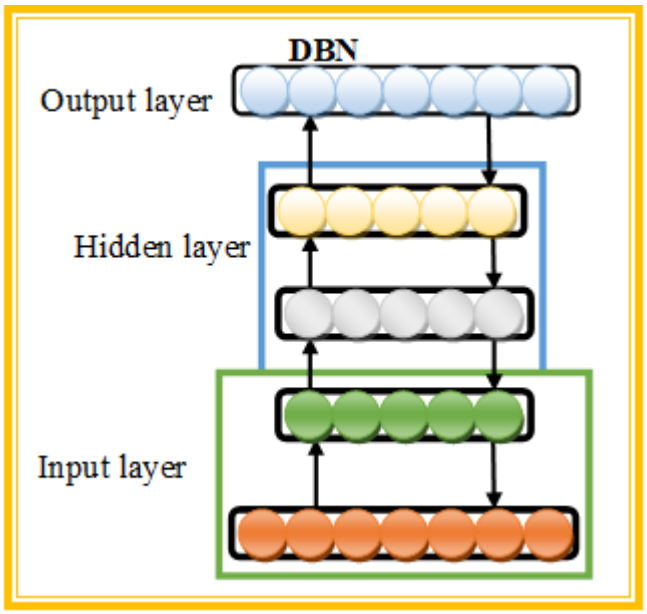

Fig. 2. Architecture view of DBN

4.2. Dilated ConvLSTM

The DConvLSTM [34] in cloud data deduplication represents a pioneering approach aimed at enhancing the efficiency and accuracy of identifying and removing duplicate data within cloud storage environments. This advanced method leverages the strengths of both CNNs and LSTM units, incorporating dilated convolutions to extend the model's receptive field without losing resolution or coverage.

A dilated convolution operation is defined by a dilation rate that dictates the spacing between the kernel elements. For example, a dilation rate of 1 means no dilation (standard convolution), a rate of 2 means that there is a space of one pixel between kernel elements, and so on. This allows the network to aggregate information over a larger spatial area, improving its ability to recognize patterns that span larger portions of the input data.

CNN excel at generating precise representations for individual data. However, capturing the progression of time requires the capabilities of a Recurrent Neural Network (RNN). ConvLSTM serves as a powerful substitute for monitoring changes across both time and space dimensions. Unlike traditional LSTM, ConvLSTM is adept at encoding variations in both time and spatial dimensions through its sophisticated gating mechanisms, offering a more detailed depiction of the data being analyzed. The ConvLSTM model is mathematically detailed through Eq. (6) to Eq. (11).

$$P_s = \sigma(k_d V_s + k_d V_{s-1} + M_P) \quad (6)$$

$$D_s = \sigma(k_d V_s + k_d V_{s-1} + M_D) \quad (7)$$

$$L_s = \sigma(k_d V_s + k_d V_{s-1} + M_L) \quad (8)$$

$$H_s = Tank(k_{dk} V_s + k_{dk} V_{s-1} + M_H) \quad (9)$$

$$C_s = L_s \otimes H_{s-1} + P_s \otimes H_s \quad (10)$$

$$W_s = C_s \otimes \tan k(C_{s-1}) \quad (11)$$

In the LSTM architecture, each memory block comprises three distinct gates: an input gate P_s, a forget gate D_s and an output gate L_s. These gates employ the sigmoid activation function denoted by σ. The biases and weights associated with these gates are represented as V_s and M_s, respectively.

The feature in the input vector at the current step is denoted by S. Element-wise duplication is depicted as $\otimes$. The memory cell state and hidden layer output at any given time are denoted as H_s and W_s, respectively.

The output from fully connected layers offers a global overview of the entire dataset but often misses out on detecting nuanced spatial variations. This shortfall leads to the use of additional data types, like optical flow data, which complicates the computational process. DConvLSTM stands out as an efficient solution by encoding the convolution features derived from CNNs. Additionally, its convolution learning gates are specifically trained to notice changes over time in certain areas, thus enabling the network to effectively capture both the detailed spatial information and the temporal dynamics. Fig. 3 shows the architecture view of DConv-LSTM.

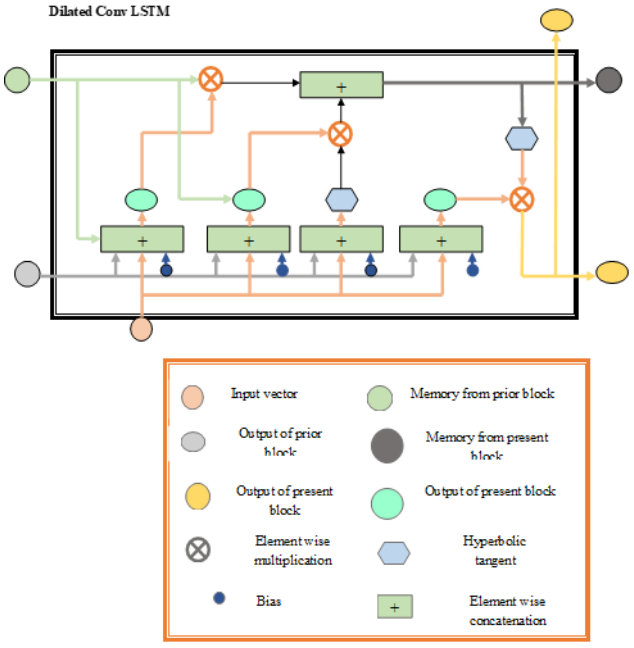

Fig. 3. Architecture view of DConv-LSTM

4.3. Proposed OSCDN for Deduplication

OSCDN is a framework designed specifically for data deduplication tasks. It employs a cascaded architecture of deep neural networks to enhance the efficiency and accuracy of the deduplication process. By optimizing the serial connections within the network, OSCDN aims to achieve superior performance in identifying and eliminating duplicate data instances, thereby reducing storage space and improving data management workflows.

In the proposed OSCDN framework, the data attributes such as filename, size, block name, size, file-type hash tag, file location, file updated date and data pattern to facilitate the deduplication process is initially inputted into a DBN model. The DBN processes the data to identify and highlight its underlying patterns and features. Following this, the features extracted by the DBN are passed in a subsequent model DConvLSTM. This model enhanced with dilated convolutions, is adept at capturing both spatial and temporal dependencies within the data and eliminating duplicates. The end result of this sequential processing through DBN and DConvLSTM is the production of deduplicated data, effectively removing redundant information.

DBNs, characterized by multiple layers of stochastic, latent variables, are generative models. The capacity of these models to learn complex data representations is influenced by the number of hidden neurons in each layer. Conv-LSTM, on the other hand, integrates convolution layers into the LSTM architecture, making it particularly suitable for processing spatial-temporal data. The hidden neuron count in Conv-LSTM impacts its capability to capture spatial hierarchies and temporal dependencies. To mitigate errors of this nature, optimization is performed using parameters such as “hidden neuron count and epoch count in both DBN and DConv-LSTM models”. This optimization aims to enhance precision through the implementation of a proposed ERTH algorithm. The goal factor of the model is represented as following the Eq. (12).

$$UY_1 = \underset{\{H_D, H_{CL}, E_D E_{CL}\}}{\arg\max} (\lambda) \tag{12}$$

From the provided information, the variables H_D and H_{CL} represent the number of hidden neurons of DBN and DConv-LSTM”, while E_D and E_{CL} denote the number of epochs of DBN and DConv-LSTM, which both range from 5 to 255 and 5 to 50, respectively. Additionally, λ specifies the precision, with its mathematical formulation estimated in Eq. (13).

$$\lambda = \frac{\text{pp}}{yy + dd} \tag{13}$$

Here, word pp and yy, dd and tt denotes “true, false positives and false, true negatives”. Fig. 4 shows the view of proposed OSCDN for data deduplication.

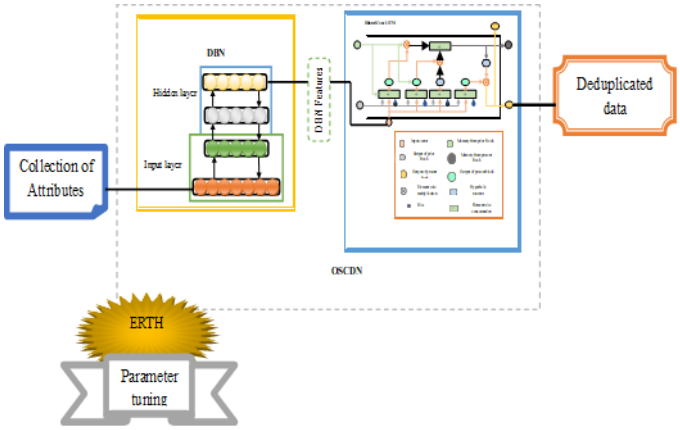

Fig. 4. View of proposed OSCDN model for data deduplication

5. Enhancement of Data Security using Improved Heuristic Algorithm based Hyper ECC Technique

5.1. Enhanced RHA

Purpose: The optimization approach ERTH leverages the principles of the RTH [26] to refine the parameters involved in data deduplication and encryption in cloud environment. This strategy

is inspired by the predatory tactics of the RTH, adept at efficiently managing the balance between exploration (broad area search) and exploitation (focused search on promising zones). This balance enables the algorithm to perform effectively in optimizing complex problems. The inherent decentralized characteristics of such natural-inspired behaviors make these algorithms well-suited for parallel computing frameworks, enhancing convergence speeds on appropriate hardware platforms. However, there's a potential downside in that the algorithm could become overly adapted to specific problem nuances or become ensnared in local optimums if the exploration-exploitation equilibrium is not properly maintained. While this method shows efficacy across various problem types, the computational demands might escalate with problem scale, rendering it less viable for very large optimization challenges.

Novelty: To counteract above mentioned challenges, the proposed methodology incorporates adjustments to diminish these drawbacks. This enhanced methodology is mathematically represented in Eq. (14).

$$\delta = \frac{Cf}{Bf + Wf} \tag{14}$$

The variables Cf, Bf and Wf mentioned in the formula represents the revised random variable, which reflects the present Cf, minimum Wf and maximum Bf fitness values. The alteration of this random variable is detailed in Eq. (20).

The red-tailed hawk is a predator with a varied diet, considering nearly any small animal it encounters as potential prey. While their diet primarily consists of small mammals like mice, they also feed on birds, fish, other vertebrates, amphibians, and invertebrates. The availability of prey differs significantly across locations and seasons, yet mice make up about 85% of a hawk's diet.

High rising: Red-tailed hawks ascend to great heights to locate the best food sources. At this juncture, Eq. (15) represents the mathematical model.

$$T(y) + T_{best} + (T_{mean} - T(y-1)).levy(\dim), LP(y) \tag{15}$$

In this model, $T(y)$ symbolizes the location of the red-tailed hawk at the y_{th} iteration, T_{best} represents the optimal position achieved, and T_{mean} denotes the mean of all positions. The Levy coefficient, indicative of the levy flight distribution, is calculated using Eq. (16), while $LP(y)$, the dynamic factor, can be determined according to Eq. (17).

$$levy(\dim) = x\frac{\delta.\mu}{|L|^{k-1}} \tag{16}$$

$$\mu = \left(\frac{\tau(1+\mu).\sin\left(\frac{\varphi\mu}{2}\right)}{\tau\left(1+\frac{\mu}{2}\right).2^{\left(\frac{1+\mu}{2}\right)}} \right) \tag{17}$$

In this context, x is set at 0.01 and represents a variable, $\dim$ refers to the dimensions of the problem, and μ is another variable fixed at 1.5. Meanwhile, δ are the random values that range between 0 and 1.The transition factor is modelled in Eq. (18)

$$LP(y) = 1 + \sin\left[\frac{l}{L_{\max}}\right] \tag{18}$$

The variable $L_{\max}$ is the highest number of iteration.

Low rising: The hawk descends closer to the ground, circling its prey in a spiral pattern, which can be described as following Eq. (19):

$$T(y) = T_{best} + (t(y) + v(y)) : stepsize(y)$$
$$stepsize(y) = T(y) - T_{mean} \tag{19}$$

In this case, t and v represent directional coordinates, which can be calculated using the subsequent Eq.(20).

$$\begin{cases} t(y) = m(y).\sin\theta(y).\delta \{ (t(y)/\max/t(y) \\ t(y) = m(y).\sin\theta(y).\delta \{ (t(y)/\max/t(y) \end{cases} \tag{20}$$

Typically, the value of a random number δ is generated through conventional methods and falls within the 0 to 1 range. Yet, this range's variability can complicate the process of achieving convergence and attaining the appropriate level of precision. To mitigate these challenges, we introduce our proposed formula, encapsulated in Eq.(14).

Kneeling and Descending (Stooping): During the phase, the hawk prepares to strike its target from an advantageous low-hovering position. The model for this stage is as follows in Eq. (21).

$$T(y) = \mu(s).T(y).stepsize1(y) + T(y).stepsize1(y) \tag{21}$$

The calculation for each dimension of a step can be carried out as following Eq. (22).

$$\begin{cases} stepsize1(y) = T(y) - LP(y).T_{mean} \\ stepsize\,2(y) = T(y) - LP(y).T_{best} \end{cases} \tag{22}$$

Like many bio-inspired algorithms, an RTH algorithm would likely excel in adaptability, capable of navigating complex landscapes to find optimal solutions, akin to how a red-tailed hawk can adapt to various environments for hunting. Fig 5 shows the proposed approach's flowchart.

Algorithm 1: Developed ERTH			
Begin			
Define the constraints and variables			
Set up execution parameters			
Initiate the population matrix and define the objective function.			
Find the better position			
While $L > L_{\max}$			
	Upgrade the random value δ using Eq.(14)		
		High Raising	
			The flight distribution is calculated using Eq.(16)
			Transition factor LP is estimated using Eq.(18)
			The position is updated using Eq.(17)
		Low Raising	
			The direction co-ordinates are estimated using Eq.(19)
			The position is updated using Eq.(20)

		Kneeling and Descending (Stooping) phase:	
			The step size is estimated using Eq.(21)
			The position is updated using Eq.(22)
		End	
End while			
The better position is obtained			
The best position is stored			

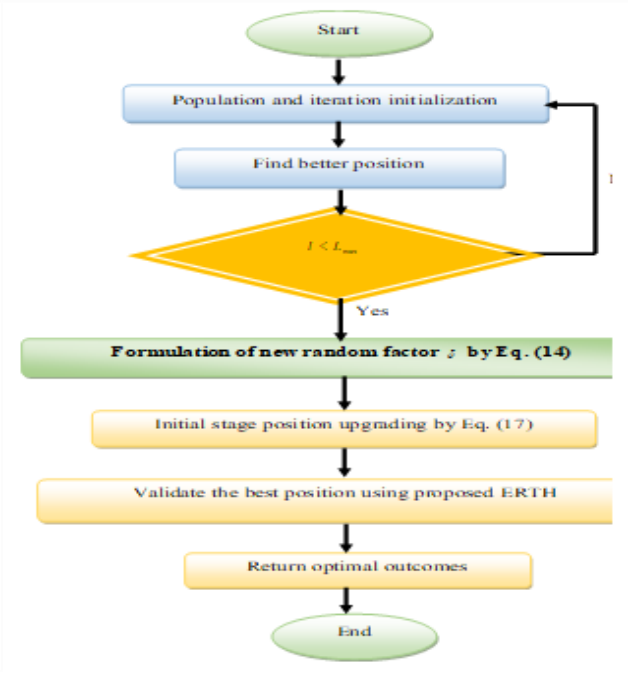

Fig. 5. Proposed system's flowchart

5.2. Hyper ECC

Hyperelliptic curves [32], akin to elliptic curves, serve as robust options for encryption purposes. To enhance the deduplication process, encryption is employed to prevent unauthorized user access. The equation representing a hyperelliptic curve, which delineates the curve's genus s across F the specified field M , is as following Eq. (23):

$$M = s_2 + F(s)E = u(s) \tag{23}$$

The methods developed leverage these properties to facilitate file encryption on the user's end using HECC.

Key agreement: HECC is a form of asymmetric cryptography that utilizes a pair of keys: a public key and a private key unique to each user. The public key, which is openly shared, is used for encrypting data and verifying signatures.

In the client application, the HECC technique is employed to create a pair of keys $\{pr, s\}$: a private key s and a public key pr comprising what is known as a key pair. The concept of key pairing integrates both these keys. While the additive group of integers is originally tailored for the Diffie-Hellman key exchange protocol, this concept is versatile and can be adapted for use with more general group structures. Imagine a scenario e is group whose elements are straightforward to define and evaluate. Such a group is formed from the Jacobian of hyperelliptic curves.

These public variables are taken into consideration:

- The group s .
- An element s of the group with a large prime order $B \in s$.

Encryption/Decryption: Before uploading files to the cloud, the owner of the information secures them through encryption using a public key $pr--h$. For additional verification later on, the hash of the information is calculated and kept. The encrypted data is then uploaded to the cloud.

When a data user wishes to access a file, k send a request to download the file from the cloud L to N . The file is then decrypted using a decryption key. Upon accessing the content, the hash value is recalculated. Through comparison, the integrity of the file is verified. By comparing the recalculated hash value with the one stored in the cloud, it becomes possible to confirm if the file has remained unaltered during its storage in the cloud.

Signature Schemes: The electronic signature technique enables the creation and verification of signatures for any group k . To sign a message L , the sender must undertake the following steps:

- Compute $Z = EB$ using a randomly chosen number $m \in [1, s-1]$.
- The resultant signature is given by (s, Z, L) .
- Compute X_1 and X_2 based on $D(L)$ and Z .
- Calculate the verification parameter $X = X_1 B + X_2 p$.
- If $X = Z$, the signature is deemed valid and thus accepted. Otherwise, it is rejected. Finally, the encrypted data is obtained.

Fig. 6 shows the architecture view of HECC.

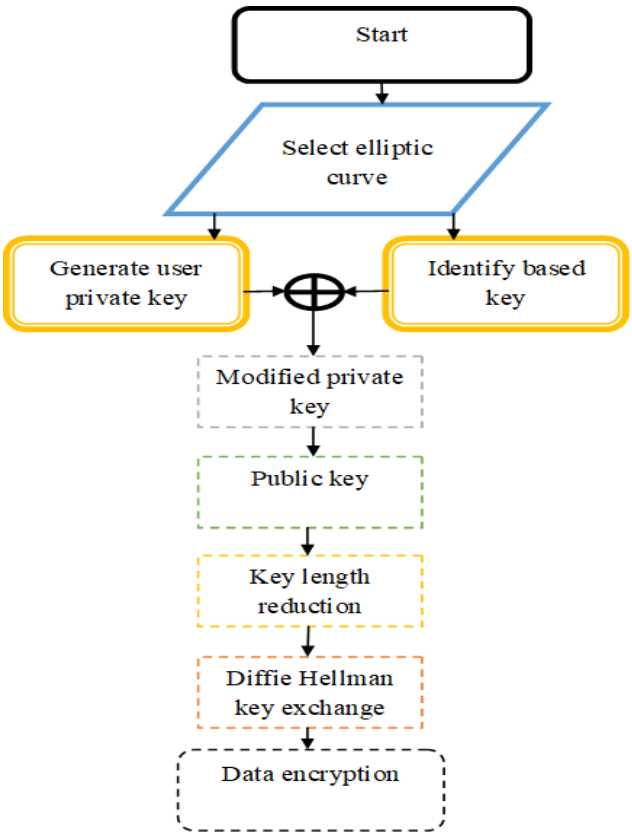

Fig. 6. Architecture view of HECC

Suggested HECC-OK for Security Purpose

In today's data-driven world, ensuring the security and privacy of sensitive information is paramount. One significant challenge in this regard is managing large volumes of data efficiently while maintaining robust security measures. To address this challenge, a cutting-edge approach known as HECC-OK is being employed. HECC-OK combines the power of HECC with the concept of optimal key management. This innovative technique is designed to encrypt de-duplicated data effectively, providing enhanced security without compromising on performance. Without proper optimization, ECC systems might encounter enhanced computational demands. Although binary keys can offer computational simplicity because of their representation, meticulous management is essential to dodge inefficiencies during encryption and decryption activities. To alleviate these challenges, ensuring minimal consumption of time and memory, variables like key in binary format for ECC undergo optimal tuning with the help of proposed ERTH algorithm. Eq. (24) outlines the mathematical expression for the objective function of the proposed system.

$$UY_2 = \arg\min_{\{O_{key}\}} (T + M) \tag{24}$$

From the provided information, the variables O_{key} represent the optimal key and the range is 0 or 1, while T and M denote the time and memory respectively. Fig. 7 shows the proposed view of Suggested HECC-OK for encryption.

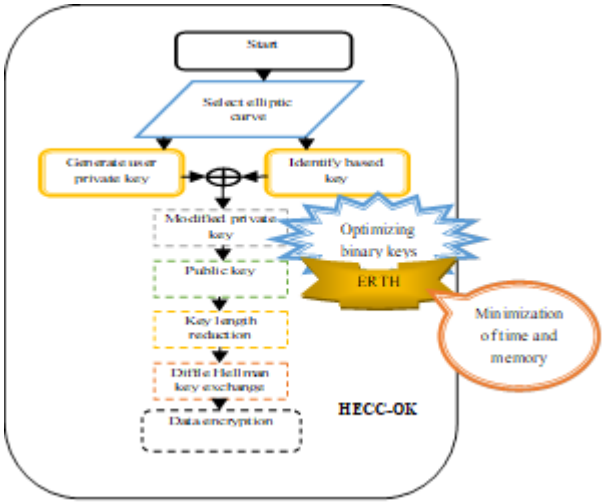

Fig. 7. Proposed view of suggested HECC-OK for encryption

6. Results and Discussions

6.1. Experimental setup

The creation of the data deduplication framework, significantly enhanced by Python programming, yielded impressive outcomes. Utilizing the ERTH method, which operated over 50 iterations for a population of ten individuals and incorporated 4 chromosomes, this framework demonstrated its efficacy. In conjunction with this, the model employed cutting-edge optimization algorithms including the "Golden Eagle Optimizer (GEO)-OSCDN [27], Red Fox Optimization (RFO)-OSCDN [28], Chef-based Optimization Algorithm (CBOA)-OSCDN [29] and the RTH-OSCDN [26] to optimize the performance. To comprehensively assess its effectiveness, the model was benchmarked against established cryptographic standards such as Data Encryption Standard (DES) [30], Attribute-Based Encryption (ABE) [2], Advanced Encryption Standard (AES) [31] and Hyper-ECC [32], providing a thorough comparison of its capabilities.

6.2. Evaluation measures

The implemented data deduplication framework uses a number of efficacy metrics and it is given below.

(a) Accuracy: $$acy = \frac{(pp + yy)}{(pp + yy + dd + tt)}$$

(b) FNR: $$fnr = \frac{pp}{dd + tt}$$

(c) FPR: $$fpr = \frac{yy}{pp + yy}$$

(d) Specificity: $$SPE = \frac{pp}{dd + tt}$$

(e) Sensitivity: $$SEN = \frac{yy}{pp + tt}$$

(f) NPV: $$NPV = \frac{pp}{tt + dd}$$

Here, word pp and yy, dd and tt denotes true, false positives and false, true negatives.

6.3. Convergence experiment of the suggested ERTH algorithm

Fig. 7 illustrates the comparative analysis of the convergence efficiency of the newly introduced ERTH method against conventional methods by examining iteration values. Specifically, at the 20th iteration, the ERTH method exhibited superior convergence rates, surpassing other methods significantly: it outperformed the GEO by 65.9%, the RFO by 34.5%, CBOA by 89% and the RTH algorithm by 29%. These results underscore the effectiveness of the modified ERTH approach in optimization tasks compared to existing strategies.

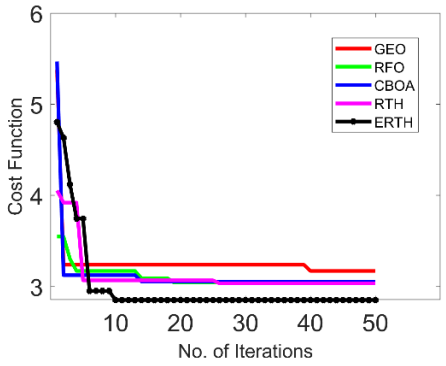

Fig. 8. The designed ERTH algorithm's convergence evaluation over multiple traditional optimization algorithms

6.4. Block size analysis of the proposed cryptograpy techniques compared over existing model

The proposed HECC-OK based encryption model underwent comparative evaluation with several cryptographic techniques, as depicted in Fig. 9. This evaluation considered various block sizes to assess the performance comprehensively. Particularly, in Fig. 9(b), where the block size memory was fixed at 20, the cryptography algorithm demonstrated remarkable superiority,

outshining DES, ABE, AES an HECC by margins of 33.9%, 77.5%, 34% and 22%, respectively. Such outcomes highlight the exceptional efficacy of the proposed model, showcasing its advanced capabilities beyond traditional classification methodologies.

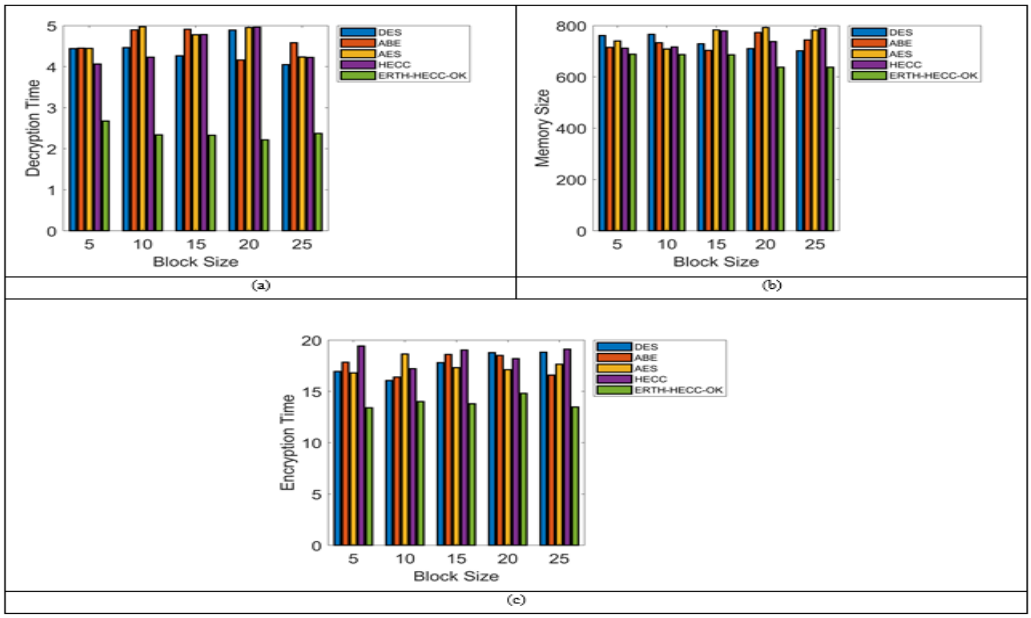

Fig. 9. Block size analysis of the designed HECC-OK based encryption model contrast with traditional cryptograpy algorithms regarding "(a) Decryption time, (b) Memory size and (c) Encryption time "

6.5. Block size analysis of the proposed ERTH- HECC-OK model compared over existing model

The proposed HECC-OK based encryption model subjected to a comparative analysis alongside several traditional algorithms, illustrated in Fig.10. This analysis took into account different block sizes to thoroughly evaluate performance. Notably, in Fig. 10(c), with the encryption time set at 25, the proposed algorithm significantly outperformed GEO-HECC-OK, RFO-HECC-OK, CBOA-HECC-OK, and RTH-HECC-OK, surpassing them by margins of 22.9%, 56.5%, 78%, and 55% respectively. These results underscore the outstanding effectiveness of the proposed model, highlighting its superior capabilities over conventional data deduplication methodologies.

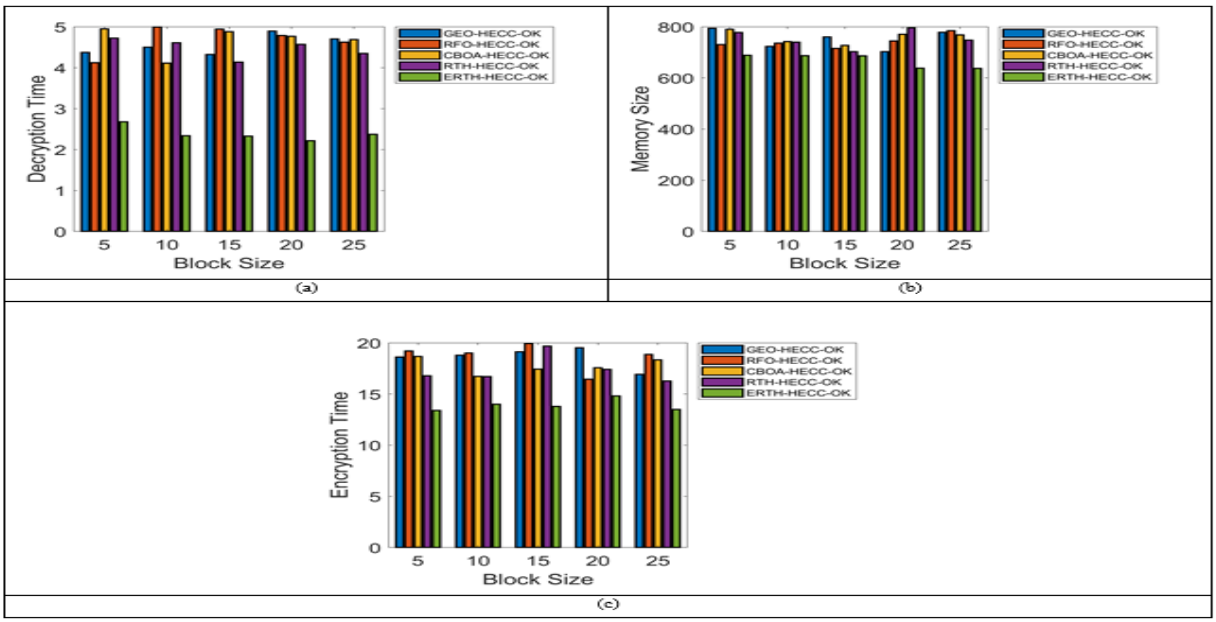

Fig. 10. Block size analysis of the designed HECC-OK based encryption model contrast with traditional algorithms regarding "(a) Decryption time, (b) Memory size and (c)Encryption time "

6.6. Case based analysis of the proposed HECC-OK based encryption model compared over conventional model

The analysis of the HECC-OK based encryption model involved a comparative review against various established algorithms, as depicted in Fig.11. This review considered a spectrum of case-specific variables. Specifically, in Fig. 11 (a), with the CPA ratio is set as 2, the proposed model demonstrated superior performance, surpassing GEO-HECC-OK, RFO-HECC-OK, CBOA-HECC-OK, and RTH-HECC-OK by margins of 44.9%, 66.5%, 89%, and 11%, respectively. These findings emphasize the enhanced effectiveness of the proposed model compared to other traditional algorithm models.

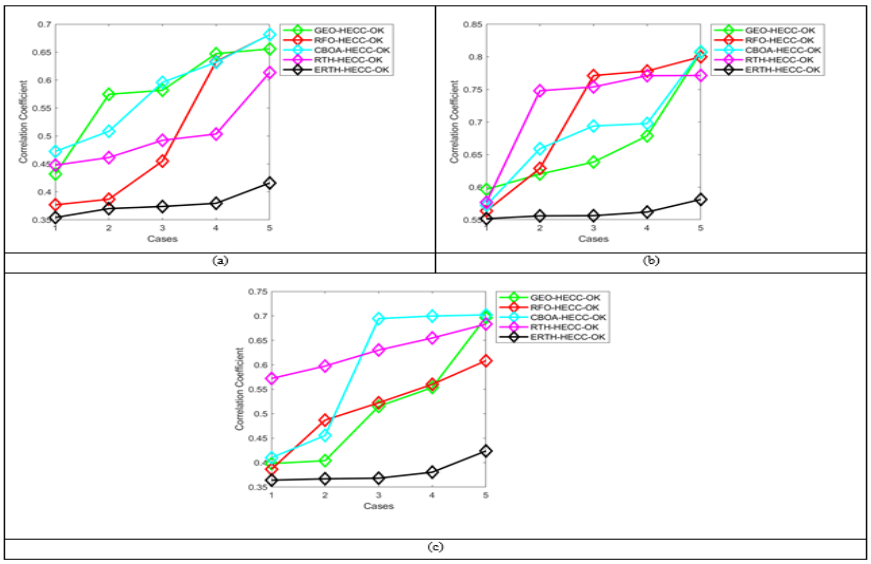

Fig. 11. Case based analysis of the designed HECC-OK based encryption model contrast with traditional algorithms regarding "(a) CPA, (b)Key sensitivity and (c)KPA"

6.7. Case analysis of the proposed cryptography technique model compared over conventional model

The evaluation of the HECC-OK based encryption model involved a detailed comparison with several recognized cryptography techniques, as shown in Fig. 12. This examination took into account a range of factors specific to each case. Notably, in Fig. 12 (b), when the key sensitivity is set as 4, the performance of the proposed model notably outperformed that of DES, ABE, AES, and HECC, with improvement margins of 78.9%, 56.5%, 77%, and 12%, respectively. These results underscore the superior efficacy of the proposed model in comparison to traditional cryptographic algorithms.

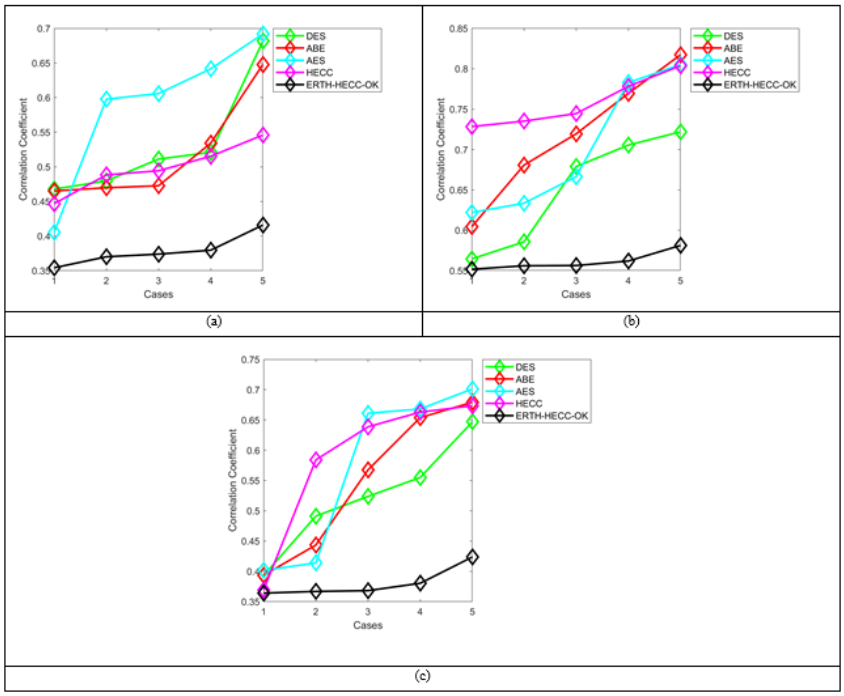

Fig. 12. Case based analysis of the designed HECC-OK based encryption model contrast with traditional cryptography algorithms regarding "(a) CPA, (b)Key sensitivity and (c)KPA"

6.8. Performance analysis of the data deduplication model compared over conventional approaches

The proposed data deduplication method underwent comparison with a variety of conventional approaches, as illustrated in Fig. 13. This evaluation specifically considered the impact of different activation function variables. Notably, in Fig. 13(a), when the activation function's accuracy was set to use the tanh function, the proposed model demonstrated superior performance, exceeding that of GEO-OSCDN, RFO-OSCDN, CBOA-OSCDN, and RTH-OSCDN by margins of 33.9%, 6.5%, 64%, and 22%, respectively. These results underline the remarkable effectiveness of the proposed model, showcasing its advantage over conventional classification techniques.

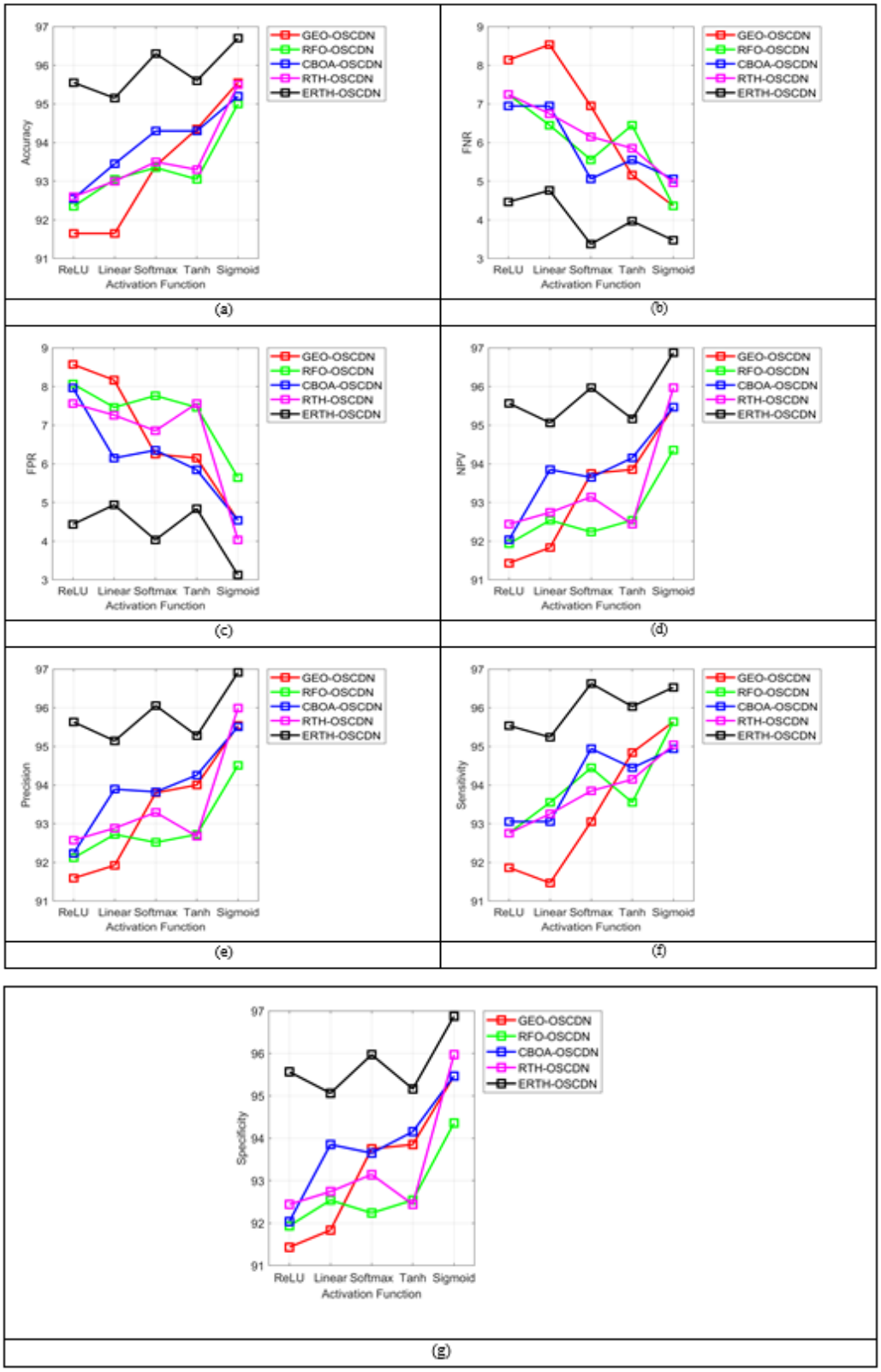

Fig. 13. Performance analysis of the designed data deduplication in cloud network contrast with traditional algorithms regarding (a)accuracy,(b)FNR,(c)FPR,(d)NPV,(e)Precision,(f)sensitivity and (g) specificity"

6.9. Overall Performance Analysis of the data deduplication model

Tables II and III provide a comprehensive comparison of the efficiency of the proposed model with other traditional algorithms. Notably, the suggested data deduplication model for the cloud exhibited precision rates surpassing 22.9% for GEO-OSCDN, 78.5% for RFO-OSCDN, 67% for CBOA-OSCDN, and 49% for ERTH-OSCDN. These findings underscore the remarkable accuracy and overall effectiveness of the recommended approach.

Table II: The Implemented Data Deduplication in Cloud Network Compared Over Multiple Conventional Algorithms

TERMS	GEO-OSCDN [27]	RFO-OSCDN [28]	CBOA-OSCDN [29]	RTH--OSCDN [26]	ERTH-OSCDN
"Accuracy"	95.55	95	95.2	95.5	96.7
"Sensitivity"	95.635	95.635	94.94	95.04	96.528
"Specificity"	95.464	94.355	95.464	95.968	96.875
"Precision"	95.54	94.51	95.509	95.992	96.912
"FPR"	4.5363	5.6452	4.5363	4.0323	3.125
"FNR"	4.3651	4.3651	5.0595	4.9603	3.4722
"NPV"	95.464	94.355	95.464	95.968	96.875
"FDR"	4.4599	5.4902	4.491	4.008	3.0876
"F1-score"	95.588	95.069	95.224	95.513	96.72
"MCC"	91.099	90.005	90.401	91.005	93.401

Table III: The Implemented Hecc-ok Based Encryption Model Compared Over Multiple Conventional Cryptgraphy Techniques

TERMS	DES [30]	ABE [2]	AES [31]	HECC [32]	ERTH-OSCDN
"Accuracy"	95.55	95.35	95.25	96.55	97.3
"Sensitivity"	95.984	95.299	95.69	96.18	97.258
"Specificity"	95.097	95.403	94.791	96.936	97.344
"Precision"	95.331	95.58	95.039	97.036	97.448
"FPR"	4.903	4.5965	5.2094	3.0644	2.6558
"FNR"	4.0157	4.7013	4.3095	3.8198	2.7424
"NPV"	95.097	95.403	94.791	96.936	97.344
"FDR"	4.6693	4.4204	4.9611	2.9644	2.5515
"F1-score"	95.656	95.439	95.364	96.606	97.353
"MCC"	91.097	90.697	90.497	93.102	94.598

7. Conclusion

The proposed framework addressed the challenge of managing rapidly growing data volumes in cloud environments by introducing a sophisticated deep learning-based data deduplication model, specifically designed to optimize storage efficiency and minimize data redundancy. This model enhanced by the ERTH algorithm, aimed to improve security and reduce computational overhead. At the heart of this framework was the OSCDN, which integrated the DBN with DConvLSTM networks. This integration was crucial for effectively identifying and eliminating duplicate data, with the ERTH algorithm fine-tuning the OSCDN model's parameters to ensure peak performance. Additionally, the framework incorporated the HECC-OK model for encryption, delivering robust security solutions optimized for efficiency in resource-limited settings. The ERTH algorithm also played a role in selecting optimal keys for this encryption model, further enhancing its effectiveness. To ascertain the effectiveness and efficiency of the developed framework, it was rigorously compared against traditional data deduplication and encryption methods. This comparison utilized statistical analysis to demonstrate the proposed framework's improvements over existing techniques. When the activation function's precision was set to use the linear function, the proposed model demonstrated superior performance, exceeding that of GEO-OSCDN, RFO-OSCDN, CBOA-OSCDN, and RTH-OSCDN by margins of 55.9%, 22.5%, 83.4%, and 56.9%, respectively. These results underlined the remarkable effectiveness of the proposed model, showcasing its advantage over conventional classification techniques. The deduplication process can add complexity to data management, making it difficult to ensure consistency and integrity across the system. While data deduplication in the cloud offers significant benefits in terms of efficiency and cost savings, addressing its limitations requires ongoing technological advancements.

References

[1] Luo, Shengmei, Guangyan Zhang, Chengwen Wu, Samee U. Khan, and Keqin Li, "Boafft: Distributed deduplication for big data storage in the cloud," IEEE transactions on cloud computing, vol. 8, no. 4, pp.1199-1211, 2015.

[2] Yan, Zheng, Mingjun Wang, Yuxiang Li, and Athanasios V. Vasilakos, "Encrypted data management with deduplication in cloud computing," IEEE Cloud Computing, vol. 3, no. 2, pp. 28-35, 2016.

[3] B. Mao, H. Jiang, S. Wu and L. Tian, "Leveraging Data Deduplication to Improve the Performance of Primary Storage Systems in the Cloud," IEEE Transactions on

Computers, vol. 65, no. 6, pp. 1775-1788, 1 June 2016.
[4] Sharma, Shivi, and Hemraj Saini, "Fog assisted task allocation and secure deduplication using 2FBO2 and MoWo in cluster-based industrial IoT (IIoT)," Computer Communications, vol. 152, pp. 187-199, 2020.
[5] Muthunagai, S. U, and R. Anitha, "CTS-IIoT: Computation of Time Series Data During Index Based De-duplication of Industrial IoT (IIoT) Data in Cloud Environment," Wireless Personal Communications, vol. 129, no. 1, pp. 433-453, 2023.
[6] Oladayo Olufemi Olakanmi, Kehinde Oluwasesan Odeyemi, "Faster and efficient cloud-server-aided data de-duplication scheme with an authenticated key agreement for Industrial Internet-of-Things," Internet of Things, Vol. 14, No. 100376, June 2021.
[7] Muthunagai, S. U, and R. Anitha, "TDOPS: Time series based deduplication and optimal data placement strategy for IIoT in cloud environment," Journal of Intelligent & Fuzzy Systems, vol.43, no. 1, pp. 1583-1597, 2022.
[8] Vignesh, R, and J. Preethi, "Secure Data Deduplication System with Efficient and Reliable Multi-Key Management in Cloud Storage," Journal of Internet Technology, vol. 23, no. 4, pp. 811-825, 2022.
[9] Yang, Xue, Rongxing Lu, Jun Shao, Xiaohu Tang, and Ali A. Ghorbani, "Achieving efficient and privacy-preserving multi-domain big data deduplication in cloud," IEEE Transactions on Services Computing, vol. 14, no. 5, pp.1292-1305, 2018.
[10] Gao, Yuan, Liquan Chen, Jinguang Han, Ge Wu, and Suhui Liu, "Similarity-based deduplication and secure auditing in IoT decentralized storage," Journal of Systems Architecture, vol. 142, pp. 102961, 2023.
[11] Prathima, Ch, Naresh Babu Muppalaneni, and K. G. Kharade, "Deduplication of IoT data in cloud storage," Machine Learning and Internet of Things for Societal Issues, pp. 147-157. Springer Nature Singapore, 2022.
[12] Yoosuf, Mohamed Sirajudeen, and R. Anitha, "Low latency fog-centric deduplication approach to reduce IoT healthcare data redundancy," Wireless Personal Communications, vol. 126, no. 1, pp. 421-443, 2022.
[13] A. Vijayakumar and Dr. A. Nisha Jebaseeli, "Pioneer approach data deduplication to remove redundant data from cloud storage," International Journal of Advanced Research in Engineering and Technology (IJARET), vol. 11, No. 10, pp. 535-544, October 2020.
[14] Patra, Sudhansu Shekhar, Sudarson Jena, Jnyana Ranjan Mohanty, and Mahendra Kumar Gourisaria. "DedupCloud: an optimized efficient virtual machine deduplication algorithm in cloud computing environment." In Data deduplication approaches, pp. 281-306. Academic Press, 2021.
[15] Kumar, Anil, and C. P. Shantala. "An extensive research survey on data integrity and deduplication towards privacy in cloud storage." International Journal of Electrical and Computer Engineering 10, no. 2, 2020.
[16] Periasamy, J. K., and B. Latha. "Efficient hash function based duplication detection algorithm for data Deduplication deduction and reduction." Concurrency and Computation: Practice and Experience, vol. 33, no. 3, 2021.
[17] Vignesh, R., and J. Preethi. "Secure Data Deduplication System with Efficient and Reliable Multi-Key Management in Cloud Storage." Journal of Internet Technology, vol. 23, no. 4, Pp. 811-825, 2022.
[18] Gao, Yuan, Hequn Xian, and Aimin Yu, "Secure data deduplication for Internet-of-things sensor networks based on threshold dynamic adjustment," International Journal of Distributed Sensor Networks, vol. 16, no. 3, pp. 1550147720911003, 2020.
[19] Fu, Yinjin, Nong Xiao, Hong Jiang, Guyu Hu, and Weiwei Chen, "Application-aware big data deduplication in cloud environment," IEEE transactions on cloud computing, vol. 7, no. 4, pp. 921-934, 2017.
[20] Batham, Surabhi, Ritu Prasad, Praneet Saurabh, and Bhupendra Verma, "A new approach for data security using deduplication over cloud data storage," Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT), pp. 26-27, 2018.
[21] PG, Shynu, Nadesh RK, Varun G. Menon, Venu P, Mahdi Abbasi, and Mohammad R. Khosravi, "A secure data deduplication system for integrated cloud-edge networks," Journal of Cloud Computing, vol. 9, no. 1,pp. 61, 2020.
[22] Hurst, Aaron, Daniel E. Lucani, and Qi Zhang, "GreedyGD: Enhanced generalized deduplication for direct analytics in IoT," IEEE Transactions on Industrial Informatics, 2024.
[23] Wu, Zeng, Hui Huang, Yuping Zhou, and Chenhuang Wu, "A secure and efficient data deduplication framework for the internet of things via edge computing and blockchain," Connection Science, vol 34, no. 1, pp. 1999-2025, 2022.
[24] Muthunagai, S. U., and R. Anitha, "Computing Time Series Data During Index Based De-Duplication of Industrial IoT Data in Cloud Environment," 2021.
[25] Yang, Ye, Xiaofang Li, Dongjie Zhu, Hao Hu, Haiwen Du, Yundong Sun, Weiguo Tian, Yansong Wang, Ning Cao, and Gregory MP O'Hare, "A resource-constrained edge IoT device data-deduplication method with dynamic asymmetric maximum," Intelligent Automation & Soft Computing, vol.30, no. 2, pp. 481-494, 2021.
[26] Seydali Ferahtia, Azeddine Houari, Hegazy Rezk, Ali Djerioui, Mohamed Machmoum, Saad Motahhir & Mourad Ait-Ahmed, "Red-tailed hawk algorithm for numerical optimization and real-world problems, "Scientific Reports, vol. 13, no. 12950, 2023.
[27] 1.Abdolkarim Mohammadi-Balani, Mahmoud Dehghan Nayeri, Adel Azar , Mohammadreza Taghizadeh-Yazdi," Golden eagle optimizer: A nature-inspired metaheuristic algorithm",Computers & Industrial Engineering,Volume 152, February 2021.
[28] Dawid Połap and Marcin Woźniak," Red fox optimization algorithm",Expert Systems with Application, Volume 166, 15 March 2021.
[29] Funda Kutlu Onay," A novel improved chef-based optimization algorithm with Gaussian random walk-based diffusion process for global optimization and engineering problems",Mathematics and Computers in Simulation,Vol. 212, October 2023.
[30] M. S. Mehmood, M. R. Shahid, A. Jamil, R. Ashraf, T. Mahmood and A. Mehmood, "A Comprehensive Literature Review of Data Encryption Techniques in Cloud Computing and IoT Environment," 2019 8th International Conference on Information and Communication Technologies (ICICT), Karachi, Pakistan, 2019.
[31] Xinmiao Zhang and K. K. Parhi, "Implementation approaches for the Advanced Encryption Standard algorithm," in IEEE Circuits and Systems Magazine, vol. 2, no. 4, pp. 24-46, 2002.
[32] P. Wanda, Selo and B. S. Hantono, "Efficient message

security based Hyper Elliptic Curve Cryptosystem (HECC) for Mobile Instant Messenger," 2014 The 1st International Conference on Information Technology, Computer, and Electrical Engineering, Semarang, Indonesia, 2014.

[33] G. E. Dahl, D. Yu, L. Deng and A. Acero, "Large vocabulary continuous speech recognition with context-dependent DBN-HMMS," 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 2011.

[34] T. N. Sainath, O. Vinyals, A. Senior and H. Sak, "Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks," 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 2015.