Integrating BiLSTM-CRF and DOKS for Enhanced AI-Based Data Encryption in Cloud-Edge Environments

Bo Li, Ting Wang*, Jingjing Hu
School of Electronic Engineering, Jiangsu Vocational College of Electronics and Information, Huaian, 223003, China
E-mail: lb170103@163.com
*Corresponding author

Keywords: cloud computing, edge computing, artificial intelligence, data encryption, data obfuscation of key streams

Received: July 23, 2025

This study addresses the challenge of secure and efficient data protection in cloud-edge collaborative environments by proposing an artificial intelligence–driven encryption framework. The method integrates a bidirectional long short-term memory network with a conditional random field model to extract sensitive information, and applies a stream encryption mechanism based on a modified Feistel structure to selectively encrypt identified entities. Experimental results show that the model achieves an accuracy of 0.99 and an F1 score of 0.94 in sensitive entity recognition. It requires only 134 milliseconds to encrypt and 183 milliseconds to decrypt one hundred kilobytes of data, and reaches ninety percent accuracy within twenty-eight training iterations. The encryption system also demonstrates strong statistical security with an average bit change rate of 45.5 percent and information entropy close to the ideal value of eight. These findings confirm that the proposed method effectively balances data confidentiality, processing efficiency, and real-time deployability for edge computing scenarios.

Povzetek: Članek predlaga ogrodje za selektivno šifriranje podatkov v okolju oblak–rob, ki združuje BiLSTM-CRF za prepoznavanje občutljivih entitet in algoritem DOKS za tokovno šifriranje. Rešitev dosega visoko točnost prepoznave, dobro varnost in nizko zakasnitev, primerno za robne naprave.

1 Introduction

Currently, with the progress of cloud computing, edge computing, the Internet of Things (IoT), Artificial Intelligence (AI), and other technologies, data transmission and processing requirements have become increasingly complex. Especially when it comes to sensitive data, ensuring data privacy and security has become an important issue in technical research and application practice. The collaborative architecture of cloud computing and edge computing has become a mainstream model to solve large-scale data storage and processing. In this architecture, plenty of sensitive data needs to be frequently exchanged and transmitted between terminals, edge nodes, and cloud servers. This makes the confidentiality and integrity issues of data during transmission even more severe. To effectively address this challenge, encryption technology, as one of the main means of protecting data privacy and security, is widely used in various communication and storage scenarios [1]. Although traditional data encryption algorithms can ensure data security to a certain extent, with the increase of data volume and the improvement of computing power, traditional algorithms have gradually exposed shortcomings in security and efficiency. Especially with the high computational complexity of traditional methods, performance bottlenecks arise when processing large-scale data, and security is often not fully guaranteed when facing complex attacks [2]. Therefore, this study proposes a data encryption method based on the Data Obfuscation of Key Streams (DOKS) algorithm, and combines Bidirectional Long Short-Term Memory (BiLSTM) and Conditional Random Field (CRF) to extract and encrypt sensitive information. This research aims to provide an efficient and Secure Data Encryption (SDE) scheme, which is suitable for the encryption protection of large-scale sensitive data in the collaborative environment of cloud computing and edge computing. The unique contribution of this study lies in integrating mature natural language processing models with adaptive key stream encryption mechanisms into a cohesive pipeline for protecting sensitive text data. This method prioritizes deployment feasibility, context awareness, and selective encryption that have not been fully explored in existing literature, mainly focusing on image or sensor data encryption. The introduction of DOKS effectively enhances the unpredictability and collision resistance of the key, further improving the security of the encryption process.

The innovation of the research lies in the integration of a deep learning-based named entity recognition model with a dynamic stream encryption algorithm enabling selective, context-aware encryption of sensitive text data in edge-cloud collaborative environments.Unlike conventional encryption schemes that apply uniform encryption to entire datasets, this approach identifies and encrypts only semantically relevant entities, improving both processing efficiency and privacy granularity. The architecture is lightweight, interpretable, and deployable on resource-constrained edge nodes.

2 Related works

The wide application of cloud computing and edge computing makes it particularly important to ensure the security of data during transmission and storage. Aldabbas et al. observed that Software Defined Networking (SDN) controllers generate new flow rules for each communication instance, resulting in unwanted packet overhead and delays due to inefficient bandwidth management. To address this issue, the authors proposed a bandwidth management technique based on the Random Forest (RF) algorithm, capable of calculating and predicting bandwidth usage across different applications. Experimental results demonstrated that the proposed method significantly reduced application latency [3]. Sun et al. noted the increasing integration of IoT and cloud computing technologies into smart factory environments. Given the prevalence of critical industrial data, the authors proposed a hierarchical risk assessment model for industrial IoT cloud platforms using the analytic hierarchy process, enabling platforms to self-evaluate their security posture. Experimental findings showed that the model effectively mitigated external data attack risks [4]. Liu D et al. identified that large data volumes in industrial IoT systems contributed to significant network latency. To resolve this issue, the research team proposed a data compression algorithm based on edge computing and subsequently introduced an outlier detection algorithm using the Isolation Forest technique. This approach accurately detected both gradual drifts and abrupt anomalies. Latency analysis indicated that the method could adaptively adjust device actions to satisfy control constraints [5].

Kumar R et al. have introduced a trusted privacy protection framework for industrial IoT, which is built on a deep blockchain architecture to overcome the limitations of traditional security mechanisms. The framework integrated a trust management module, a two-level privacy protection module, and an anomaly detection module. The evaluation results showed that the framework outperformed peer-to-peer privacy-preserving intrusion detection strategies on transformed datasets, achieving an accuracy of 98.97% and a detection rate of 93.87% [6]. Wu H et al. highlighted that wireless mobile edge computing could extend the computational capabilities of low-power devices in industrial IoT environments. To tackle existing challenges, the authors proposed an online optimization algorithm designed to maximize long-term system utility while maintaining a balance between throughput and fairness. Experimental validation confirmed the superior performance of the proposed method [7]. Yang L et al. identified substantial security concerns in cross-domain device data sharing within heterogeneous industrial IoT networks. Traditional cloud-based solutions were often inefficient due to centralized architectures. In response, the team proposed a blockchain-based data sharing framework. The experimental results demonstrated that this solution alleviated system load, reduced communication latency, and enhanced the efficiency of inter-domain data sharing [8]. Gilmolk et al. proposed a new lightweight chaotic encryption method. This method adopted fuzzy access control to encrypt optical images in IoT, to meet the security requirements of low-cost technology and achieve more flexible and secure access to sensitive data. The research results indicated that the algorithm design combined random chaotic mapping and fuzzy logic shift. After multiple evaluations, the proposed technique has improved in uniformity, energy, contrast, NPCR, and UACI standards compared to other methods [9]. Peng et al. designed an Image Encryption (IE) system based on a chaotic hardware architecture, utilizing a multi-scroll chaotic system and the Arnold transformation as the primary source of entropy. The image was processed using a chaotic sequence, and the Arnold transformation was applied for scrambling. Laboratory findings demonstrated that the system achieved low power consumption, high operational speed, and strong encryption performance [10]. Boussif et al. proposed a novel IE method for securing Digital Imaging and Communications in Medicine (DICOM) images. This approach involved converting the image into a pixel matrix, encrypting the image blocks individually, followed by modification of the key using the Arnold transformation. Experimental results showed that the method successfully encrypted keys and achieved shorter computation times compared to conventional algorithms [11]. Wang et al. developed a chaotic IE algorithm based on a matrix semi-tensor product and a composite key. The image was divided into four segments, each processed using Arnold transformation, and then recombined to produce the final encrypted image. The results indicated that the algorithm offered higher security than other methods and was particularly effective for color IE [12]. Jain et al. addressed privacy concerns in remote healthcare systems, where digital images often contained sensitive patient data. They proposed a chaotic IE approach integrating Arnold’s Cat mapping and a 2D Logistic Sine Coupling Map. The findings revealed that the method enhanced both the randomness and robustness of encryption, ensuring improved protection of private medical information [13]. Zarebnia Mde et al. introduced a multi-stage IE method employing a chaotic system and Arnold transformation for image scrambling. The method demonstrated strong encryption performance and resistance to various attack models, effectively safeguarding user image data [14]. Hu et al. proposed a color IE algorithm based on a 3D chaotic system. This technique applied the Arnold transformation to scramble the original image and then encrypted each RGB channel using sequences generated by the chaotic system. Experimental findings showed that the pixel values in the encrypted images were uniformly distributed, significantly enhancing encryption robustness [15].

Huang et al. designed an IE algorithm based on 2D chaotic mapping and Arnold transformation, followed by obfuscation and diffusion using chaotic sequences. The results confirmed excellent encryption effectiveness and the method’s capability to protect image information [16]. Nie et al. introduced an IE algorithm that combined a hyper-chaotic system with the Advanced Encryption Standard (AES). The process began with Arnold transformation to eliminate partial blocking, followed by compression using a discrete cosine transform, and concluded with AES encryption. It was shown that the algorithm possessed high security and strong compression efficiency [17]. Li et al. developed a lightweight data encryption algorithm targeting IoT terminals and intermediate nodes to address image data transmission security. The results demonstrated that the algorithm effectively resisted image decryption attacks, such as brute-force and differential attacks, thereby securing user privacy and data integrity [18]. Yao M et al. proposed a color image compression and encryption algorithm that integrates compressive sensing, Sudoku matrices, and hyper-chaotic mapping to enhance the security of color image data while improving transmission and storage efficiency. This algorithm introduced a novel hyper-chaotic map, enhanced the beetle optimization algorithm for compression threshold selection, and incorporated a Sudoku matrix alongside a bidirectional diffusion process. The experimental results demonstrated that the algorithm achieved strong encryption performance [19]. Singh D et al. proposed a multi-layer IE scheme that improved grayscale and color image authentication techniques to address the security challenges associated with private data in network communication. The scheme combined a public key cryptosystem with two chaotic maps. The experimental results showed that the algorithm offered a large key space, exhibited a very low correlation coefficient, and was effective in resisting statistical and various brute-force attacks [20]. Wang A et al. introduced a multi-IE method based on a computer-generated Phase-Only Hologram (POH) algorithm and chaotic systems for securely encrypting multiple images. This method utilized an improved Gerchberg-Saxton algorithm to generate sub-sampled POHs and merged them through spatial segmentation multiplexing. The composite holograms were then encrypted using chaotic systems. The study showed that this method eliminated information leakage, enhanced system complexity, and demonstrated both security and practical feasibility [21]. Liu Y et al. proposed a novel 3D medical image encoding scheme that employed biometric keys and cube structures to overcome the limitations in existing 3D medical IE techniques. This approach enhanced data security using biometric authentication and increases nonlinearity through the use of cube structures. The experimental results indicated that the scheme offered favorable statistical properties, a large key space, high sensitivity, and robustness, effectively resisting various typical cryptographic attacks [22].

In summary, existing research, including chaotic systems, the Arnold transform, and deep learning models, provides multiple solutions for IE, anomaly detection, and IoT terminal security. However, these methods often focus on structured data formats like images or sensor readings, and rarely address unstructured text data containing sensitive personal information in collaborative edge-cloud environments. There are still problems, such as low encryption efficiency and insufficient model generalization ability. Therefore, this study proposes an SDE method based on BiLSTM and DOKS. It extracts sensitive information through deep learning models and combines it with an improved DOKS algorithm to improve the precision and efficiency of data encryption, providing an efficient and secure technical solution for large-scale data encryption.

3 Methods

3.1 Data encryption method based on DOKS

With the emergence of 5G, IoT, AI, and other emerging technologies, the collaborative architecture of cloud computing and edge computing has become the mainstream model of data processing and service delivery. Under this architecture, numerous sensitive data need to be frequently transmitted between terminals, edge nodes, and cloud servers. Ensuring the confidentiality and integrity of data during transmission and storage is an urgent issue to be addressed [23]. An encryption algorithm is a rule and process for encrypting data. It converts data through specific algorithmic logic and keys, making it unreadable or tamper-proof without authorization. This study proposes a privacy data protection method based on cryptographic techniques. Figure1 shows the overall structure.

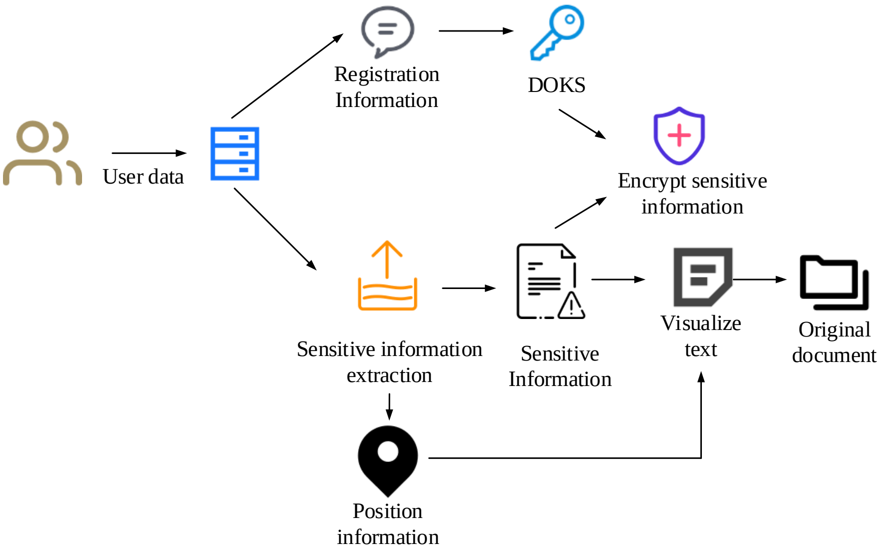


Figure 1: Overall structure of encryption method

In Figure1, the system first receives the user's registration and login data. These data include sensitive information and location information. Through the Sensitive Information Extraction (SIE) module, the system extracts sensitive information that needs to be protected from the raw data and uses it as an encrypted object. Next, the system introduces DOKS to generate a dynamic key stream based on user registration information. This key stream is combined with sensitive information for encryption operations to generate visual text and the final ciphertext file. Visual text is easy for users to understand and display, while encrypted files ensure the security of sensitive data. Throughout the process, the original file is protected after encryption, avoiding the risk of sensitive information leakage during transmission and storage. The most crucial method in this process is the generation of the key stream, whose quality directly affects the security of the system, as shown in Figure2.

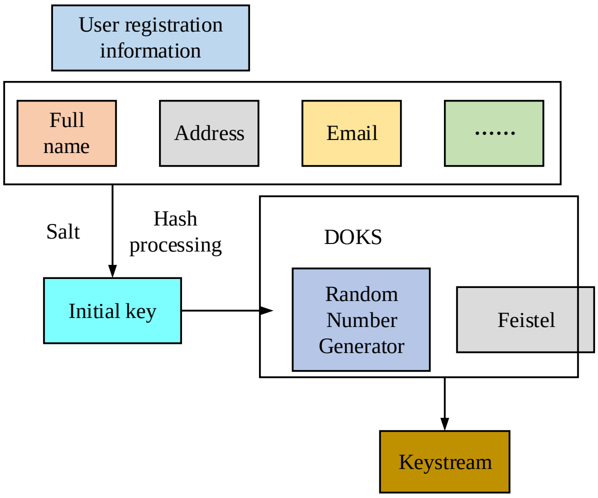


Figure 2: Key stream generation process

In Figure2, the process takes user registration information as input and enhances the uniqueness and collision resistance of the initial key by introducing random salt values for hash processing. The generation representation of the initial key is shown in equation (1).

$$K_0 = H(U,S) \tag{1}$$

In equation (1), H is the hash function, U is the user information, and K_0 is the initial key. Then, the initial key is input into the DOKS algorithm module. This module includes a Random Number Generator (RNG) and an improved Feistel structure. The RNG dynamically generates a pseudo-random sequence based on the input key, ensuring the unpredictability of the key stream [24]. The Feistel structure further introduces the idea of block encryption, achieving obfuscation enhancement of key streams through rotation functions and key mixing. The calculation of the key wheel function is displayed in equation (2).

$$K = \text{Feistel}(R, K_0) \tag{2}$$

In equation (2), R is the initial pseudo-random sequence. The Feistel structure ensures that even small input differences can significantly alter the output, improving safety. The proposed DOKS algorithm adapts a non-equilibrium Feistel architecture and integrates multi-level obfuscation and diffusion strategies to enhance practical encryption performance. The components of hash based key generation, random number injection, and reversible matrix transformation are based on standard cryptographic design principles. However, this integration aims to balance the security and efficiency of cloud edge collaboration scenarios, and enhances the integration of cyclic displacement and reversible matrix transformation through F-function enhancement modules to achieve nonlinear data diffusion. To further clarify the operational logic of the DOKS algorithm, the complete encryption process can be summarized in the following stages: user-specific registration data is collected and combined with a random salt to produce a secure initial key through hash operations; this key is input into a random number generator and a Feistel-based block structure to derive a dynamic key stream; sensitive data blocks are divided, and each half undergoes multiple rounds of non-linear obfuscation through cyclic shift, matrix transformation, and S-box substitution; the final ciphertext is obtained by combining the scrambled halves. Compared to traditional symmetric encryption, DOKS emphasizes lightweight design and dynamic key evolution, striking a balance between processing efficiency and security robustness Its mathematical expression is exhibited in equation (3).

$$F(R) = M \cdot rot(R,k) \tag{3}$$

In equation (3), M is a full rank diffusion matrix and rot is a k-bit cyclic shift operation. The non-linear layer of the G-function introduces an 8×8 composite S-box to achieve byte replacement. The key extension mechanism uses dynamic sub-key injection to nonlinearly correlate each round of encryption key with plaintext data, as shown in Figure3.

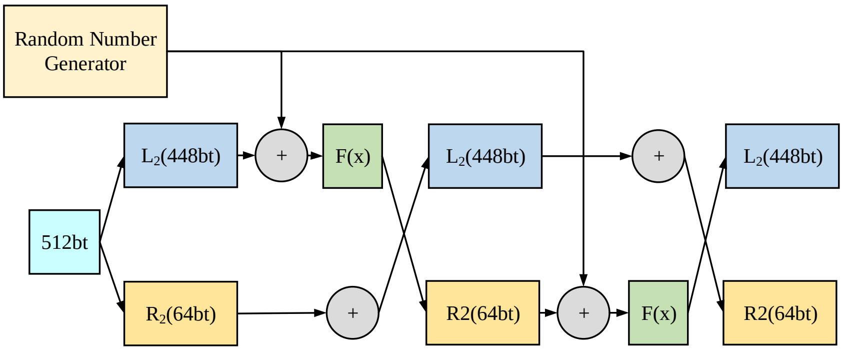


Figure 3: Asymmetric data confusion structure

In Figure3, the core processing flow of this structure is to perform multiple rounds of asymmetric scrambling on a 512-bit data block. Firstly, the input data block is divided into a left half and a right half. The RNG generates a pseudo-random number sequence to control the confusion process in each round [25]. In each round, the right half is added to the disturbance value generated in the previous round, processed through a nonlinear function, and then added to the left half to generate the right half of the next round. At the same time, the left half is shifted to the right and passed on to the next round. The disturbance value expression is shown in equation (4).

$$P_i = R_i + a_i \tag{4}$$

In equation (4), a_i comes from a pseudo-RNG, which is utilized to improve the randomness and unpredictability of each round of confusion. Next, the perturbation value is substituted into the nonlinear function and combined with the left half for updating, as shown in equation (5).

$$R_{i+1} = L_i + F(P_i) \tag{5}$$

In equation (5), $F(\cdot)$ is a nonlinear function. The entire structure utilizes an asymmetric bit partitioning method to make the left and right data processing paths different, and by continuously introducing pseudo-random factors and nonlinear functions for processing, it achieves high data obfuscation, enhances security, and anti-analysis capabilities [26]. The final output of the new round continues as input for the next round until all rounds are completed, and the final data block after confusion is output.

3.2 SDE algorithm based on sensitive information extraction

When encrypting data through DOKS-based data encryption methods, a holistic encryption approach is usually used, which lacks refined management of the data. Therefore, this study introduces the SIE module based on DOKS and employs the BiLSTM-CRF model for SIE. Although BiLSTM-CRF is a widely adopted architecture in sequence tagging tasks such as Named Entity Recognition (NER), its integration into the encryption preprocessing pipeline is adapted here to enhance entity-level control before data obfuscation. This ensures that only semantically relevant sensitive content is targeted for encryption, optimizing both efficiency and selectivity in a practical encryption context. BiLSTM can capture both forward and backward contextual information in text sequences, which is particularly important for languages such as Chinese without clear segmentation boundaries. It overcomes the limitation of traditional one-way LSTM that cannot utilize future information, thus gaining a more comprehensive understanding of the semantic relationships of words in sentences [27-28]. In practical applications, BiLSTM is particularly effective for identifying context-dependent sensitive information such as person name, address, ID number, etc. Furthermore, the CRF layer is introduced as the output layer, which not only predicts the labels of each word based on the features extracted by BiLSTM, but also considers the dependency relationships between labels. For example, in NER tasks, the model can automatically learn rules, avoiding situations where label predictions are unreasonable. Figure4 shows the structure.

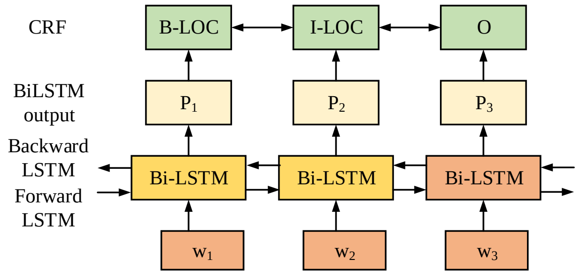


Figure 4: Sensitive information monitoring structure

In Figure4, the core of this architecture is a sequence annotation model based on BiLSTM and CRF, which is used to identify and classify sensitive entities from text. Firstly, each word in the input sequence is represented as a word vector. After entering the BiLSTM layer, both forward and backward processing will be performed simultaneously to obtain contextual information. In this framework, the BiLSTM-CRF module is responsible for fine-grained semantic extraction of sensitive content, identifying structured fields such as names, ID numbers, and addresses. These extracted entities are not encrypted as a whole document, but rather selectively processed based on their semantic relevance. The location of each sensitive field is stored separately to enable precise decryption. Once identified, these entities are handed off to the DOKS module, which encrypts each segment using a uniquely generated key stream. This modular separation between entity extraction and field-wise encryption not only improves processing efficiency but also ensures targeted protection of critical information. The output of BiLSTM is shown in equation (6).

$$h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}] \tag{6}$$

In equation (6), $\overrightarrow{h_i}$ is the hidden state generated by the forward LSTM, and $\overleftarrow{h_i}$ is the backward LSTM [29-30]. Each state will be further passed into the fully connected layer to generate a rating vector for the label space, as shown in equation (7).

$$P_i = W \cdot h_i + b \tag{7}$$

In equation (7), W means the weight matrix and b denotes the bias term. P_i contains the scoring results of the location belonging to different labels. To consider the dependency relationship between the entire label sequence, a CRF layer is introduced for joint decoding. This layer will score the entire label path, as given by equation (8).

$$s(X, y) = \sum_{i=1}^{n} P_i[y_i] + \sum_{i=1}^{n-1} A[y_i, y_{i+1}] \tag{8}$$

In equation (8), X is the input sequence, y is the label sequence, A is the label transition matrix, and y_i is the score for transitioning from the label to y_{i+1}. The ultimate goal of the model is to maximize the conditional probability of the true path, and its objective function is shown in equation (9).

$$p(y \mid X) = \frac{e^{s(X,y)}}{\sum_{\tilde{y} \in Y} e^{s(X,\tilde{y})}} \tag{9}$$

In equation (9), by training to maximize the objective function, the model can learn the optimal label combination path. Then the sensitive information is processed, and its structure is shown in Figure5.

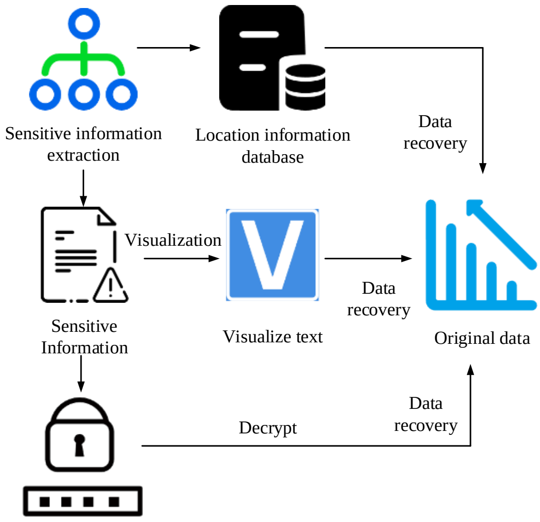


Figure 5: Framework of sensitive information processing model

In Figure5, the architecture mainly includes key components such as SIE, encryption, visualization, and data recovery. Firstly, sensitive information is extracted from the original text. This part of the data is identified and separated through the "SIE" module to form separate sensitive information, which is then stored in the location information database based on its location information. Sensitive information will be encrypted after extraction and stored in an encrypted information database to ensure data confidentiality and security. At the same time, to achieve data display and analysis, sensitive information can be transformed into visual text through visualization operations, so that users can view and process it without leaking the original sensitive content. When it is necessary to recover the original data, sensitive information is recovered from the encrypted information database through decryption operations. Based on the location data stored in the location information database, data recovery is performed and the original text is ultimately reconstructed. Finally, after encrypting the information, its structure is shown in Figure6.

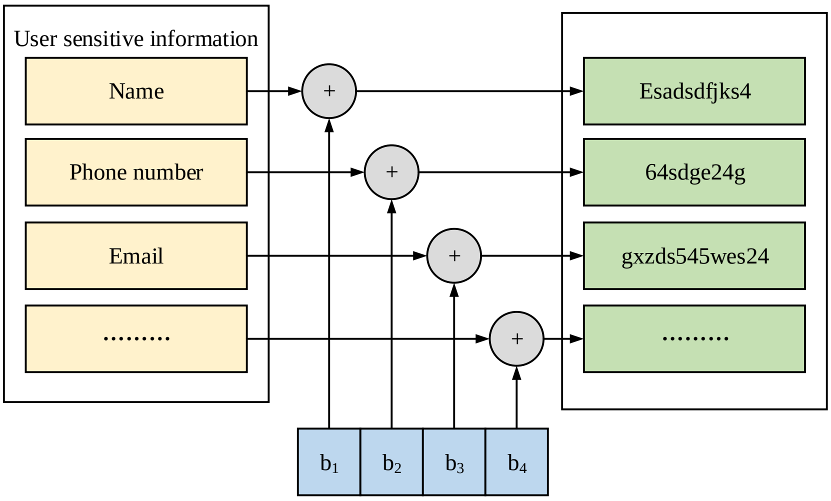


Figure 6: Sensitive information encryption process

In Figure6, first, the system extracts several sensitive fields from the user information, including name, ID number, mailbox, etc. These fields are used as plaintext input encryption modules, respectively. The DOKS algorithm is responsible for generating the key stream. Each key fragment is scrambled with its corresponding sensitive information field, that is, each sensitive data item is operated on with one or more corresponding substrings in the key stream to form the encrypted ciphertext. It is necessary to ensure that the encrypted ciphertext of each sensitive data item is completely different, even if the original text is the same, to enhance security. The final output ciphertext data is stored or transmitted, while the original sensitive information is no longer retained in plaintext form.

To enhance the replicability of the proposed method and improve understanding of the internal mechanism of key stream generation in the DOKS encryption module, the pseudocode is provided in Table 1. It illustrates how user registration information is transformed into a secure, dynamic keystream through hashing, random number generation, and nonlinear transformation.

Table 1: DOKS key stream generation process

Algorithm 1: DOKS Key Stream Generation Process

```
Input: User Info U = {name, email, address, ...}, salt (16-byte)
Output: Keystream K
$K_0 \leftarrow \text{SHA-256}(U \parallel \text{salt})$
$R \leftarrow \text{RNG}(K_0)$
for i = 1 to 6 do
    $a_i \leftarrow \text{RNG}(K_0, i)$
    $P_i \leftarrow R_i + a_i$
    $F_i \leftarrow \text{SBox}(\ \text{rot}_k(P_i) \times M\ )$
    $K \leftarrow K \oplus F_i$
end for
return K
```

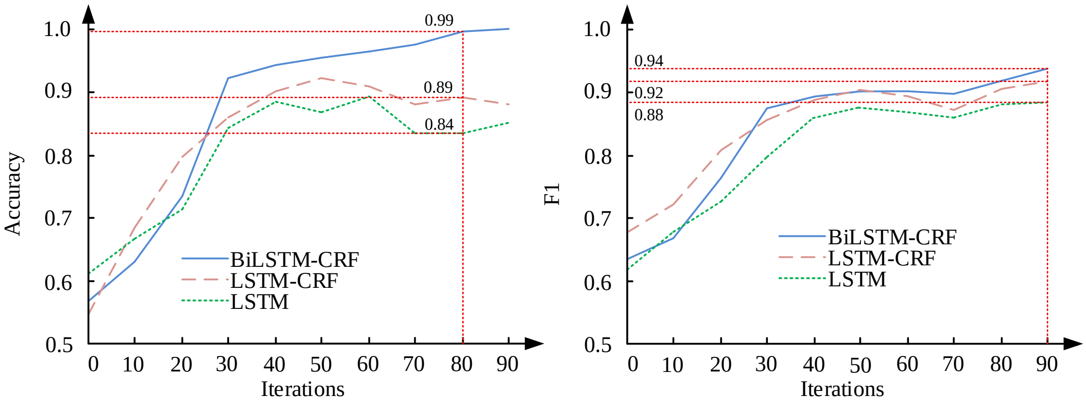
This pseudocode mirrors the modular structure introduced in the encryption method. Specifically, the initial key is generated using a SHA-256 hash of the user information and a 16-byte salt, ensuring uniqueness and collision resistance. Then, the Feistel-based scrambling process utilizes cyclic shifts, matrix multiplication, and an 8 × 8 S-box for nonlinear byte substitution. The XOR aggregation of each round enhances entropy and security.

To enhance reproducibility, the encryption and decryption steps of the DOKS algorithm are detailed as follows. The input plaintext is first divided into 512-bit blocks, which are split into left and right halves. User information combined with a 16-byte random salt (fixed seed during testing) is processed via SHA-256 to generate the initial key. A pseudo-RNG then produces round keys and disturbance values. The encryption involves 6 Feistel rounds, where in each round the right half is processed through an F-function that includes byte substitution via a fixed 8×8 S-box, cyclic right shift by i mod 8 bits, and multiplication with a full-rank 8×8 diffusion matrix. The result is XORed with the left half to update the right half, while the left half is updated directly. The ciphertext is formed by concatenating the final left and right blocks. Decryption uses the same process in reverse, with identical key regeneration and round structure, ensuring precise recovery of plaintext. All pseudo-random elements, including the diffusion matrix and S-box, are initialized with fixed seeds to ensure experimental consistency.

4 Results

4.1 Performance analysis of SDE algorithm

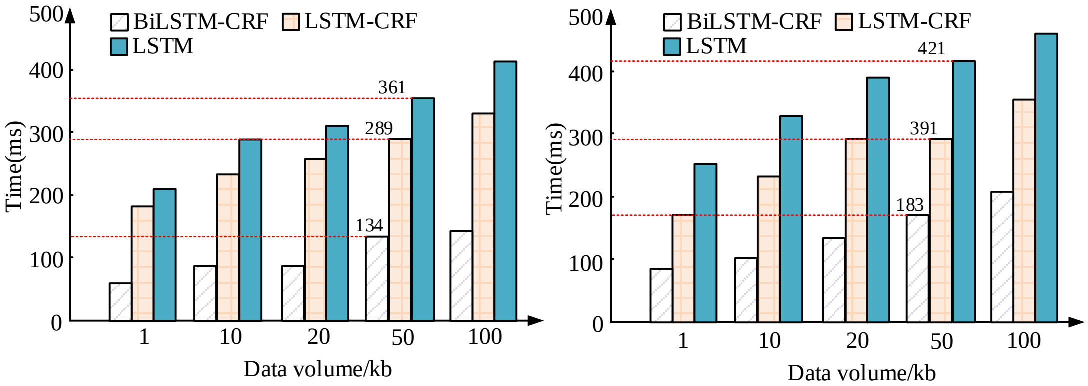
The hardware configuration uses an Intel Core i5-8750H CPU, NVIDIA GeForce GTX2080Ti GPU, 8GB of VRAM, and 16GB of RAM. The experiments are conducted on the MSRA NER dataset, which is widely used for Chinese entity recognition benchmarking. The dataset consists of over 50,000 manually annotated sentences covering diverse domains such as politics, economy, society, and sports. Entity categories include person names (PER), locations (LOC), and organizations (ORG). In preprocessing, the raw text is first segmented into individual characters as tokens, given that Chinese lacks natural word boundaries. All sequences are converted to Begin-Inside-Outside (BIO) tagging format. For consistency, all non-Chinese symbols and punctuation are removed. Sentences longer than 128 characters are truncated, and all data are lower-cased. The entire dataset is randomly divided into a training set (80%) and a validation set (20%). No external datasets are used to pre-train the models, and all models are trained from scratch under the same hyperparameter settings. This ensures a fair and controlled comparison between BiLSTM, CRF-enhanced models, and transformer baselines. In this study, different methods are used in the SIE stage of the proposed method to verify the model performances. LSTM and LSTM-CRF are selected as comparison models, as shown in Figure7.



(a) Accuracy (b) F1

Figure 7: Performance analysis of each model

Figs.7 (a) and (b) show the accuracy and F1 value changes of three models at different training iterations. In Figure7 (a), from the overall trend, the accuracy of the BiLSTM-CRF model consistently leads and reaches its highest value of 0.99 at the 80th iteration, while LSTM-CRF and LSTM remain stable at around 0.89 and 0.84. As the iteration rises, the accuracy of all models gradually improves, but LSTM fluctuates after reaching an accuracy of about 90%, indicating that its stability and generalization ability are not as good as the model with the CRF structure added. After introducing BiLSTM, BiLSTM-CRF can capture sequence information more comprehensively, and coupled with CRF's overall optimization of label sequences, the model performs better and more stably. In Figure7 (b), BiLSTM-CRF performs the best, ultimately reaching 0.94, while LSTM-CRF and LSTM are 0.92 and 0.88. Although LSTM initially rises rapidly, it tends to stabilize or even slightly decrease after 40 iterations due to overfitting issues during training. This indicates that BiLSTM-CRF, due to its bidirectional structure, enhances context understanding ability and, when combined with CRF decoding strategy, further optimizes inter-label dependencies, demonstrating stronger sequence annotation ability and overall prediction performance. An analysis is conducted on the encryption and decryption time of each model, as shown in Figure8.



(a) Encryption time

(b) Decryption time

Figure 8: Analysis of encryption time and decryption time for various models

Figs.8 (a) and (b) show the encryption and decryption times of each model under different data volumes. In Figure8 (a), as the data size increases from 1KB to 100KB, the encryption time of all models is upward, but BiLSTM-CRF always maintains the lowest encryption time. When processing 100KB of data, BiLSTM-CRF takes 134 milliseconds, while LSTM-CRF takes 289 milliseconds, with LSTM reaching a maximum of 361 milliseconds. In Figure8 (b), BiLSTM-CRF still has the shortest decryption time at various data scales, taking only 183 milliseconds for 100KB data, while LSTM-CRF is 391 milliseconds, and LSTM has the highest decryption time of 421 milliseconds. LSTM takes longer in the decryption process because it does not use CRF, resulting in scattered label prediction distributions and increased decoding complexity. Although introducing CRF into LSTM-CRF improves some efficiency, it still leads to insufficient information utilization due to the lack of bidirectional feature capture. BiLSTM-CRF, with its comprehensive feature modeling by BiLSTM and optimal path selection for output by CRF, makes the decryption process more efficient, thereby demonstrating significant performance advantages. This indicates that the proposed method has better performance. Table 2 analyzes the comprehensive performance of each model.

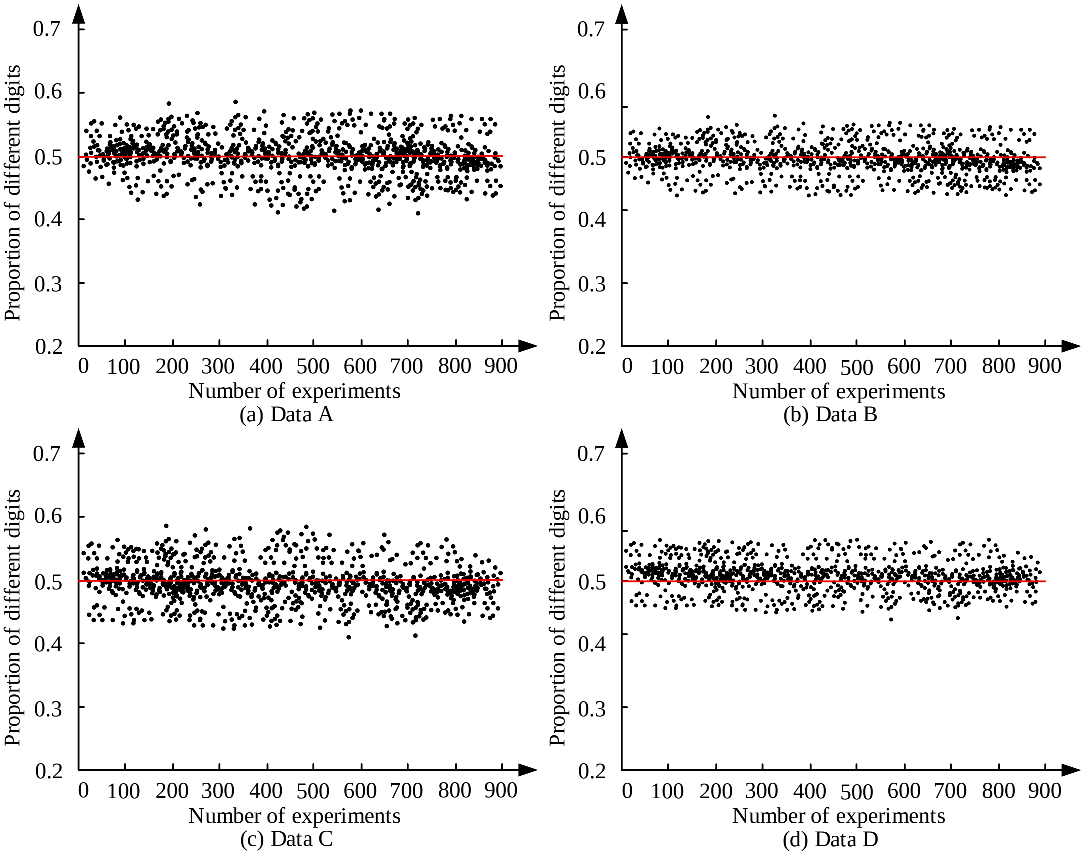
Table 2: Comprehensive performance analysis of the model

Index	BiLSTM-CRF	LSTM-CRF	LSTM
Accuracy	0.99	0.91	0.87
Recall	0.95	0.90	0.84
Precision	0.94	0.89	0.85
F1 value	0.94	0.9	0.84
Convergence wheel number	28	41	49
Average time of Encryption and decryption (ms)	158.5	295	376
Model complexity	1.5M	1.2M	0.9M
Generalization	±0.01	±0.03	±0.06

In Table 2, the BiLSTM-CRF model performs the best in various core performance indicators, with an accuracy of 0.99 and an F1 value of 0.94, far higher than the 0.90 of LSTM-CRF and 0.84 of LSTM. This indicates that it has stronger comprehensive discriminative ability in classification tasks. In terms of recall rate, BiLSTM-CRF also leads with 0.95, indicating that it can more comprehensively identify target samples and is suitable for scenarios with low tolerance for missed detections. In terms of convergence efficiency during training, BiLSTM-CRF only requires 28 rounds to achieve an accuracy of over 90%, while LSTM-CRF and LSTM require 41 and 49 rounds, reflecting their faster learning ability, which is beneficial for saving training time and computing resources. In terms of efficiency, the average encryption and decryption time of BiLSTM-CRF is 158.5 ms, much lower than the 295 ms of LSTM-CRF and 376 ms of LSTM, indicating its advantage in processing speed. In summary, BiLSTM-CRF outperforms the other two models in accuracy, convergence speed, generalization ability, and inference efficiency, making it the most balanced choice between performance and practicality.

3.2 Analysis of simulation results of SDE algorithm

To further validate encryption model's performance, this study selects four different types of data for encryption simulation, as shown in Figure9.



(a) Data A

(b) Data B

(c) Data C

(d) Data D

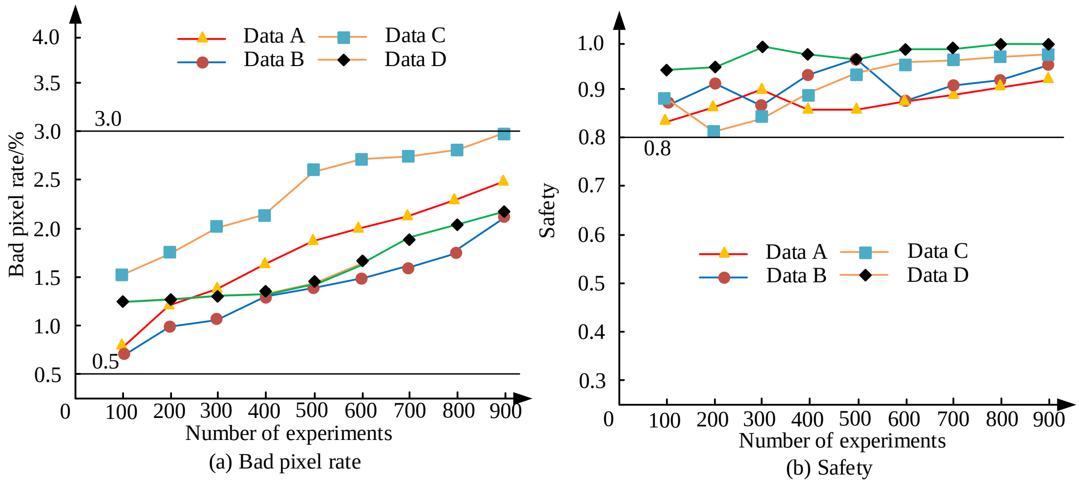
Figure 9: Analysis of avalanche effect of SDE model in different data

Figs.9 (a) to (d) show the avalanche effect test results of the encryption algorithm in different data. In the test, the original message undergoes a 1-bit change, and the bit change rate between the original and modified ciphertext, which reflects the avalanche effect, should theoretically approach 50%. In the test, the average rate is about 45%, indicating strong diffusion performance, but slightly less than ideal. This phenomenon reflects the good diffusion of encryption algorithms, which can effectively extend small changes in input data to the output. Due to the idealized range of avalanche effect being 50%, allowing for some fluctuations, the test results show that the position changes are relatively uniform, and the changes in ciphertext are not concentrated in a specific area. This indicates that the algorithm has good obfuscation and can prevent attackers from inferring based on encrypted text. Therefore, experiments have shown that the algorithm can maintain sufficient security within this range. To conduct a more statistically based evaluation of avalanche effects, this study calculates the average bit change rate of the four datasets in Table 3 after introducing a 1-bit modification in plaintext.

Table 3: Avalanche effect metrics compared with standard encryption benchmarks

Test Dataset	Average Bit Change Rate (%)	Ideal Benchmark (%)	Reference Standard	Deviation from 50% (DP, %)	LP_max
Data A	45.2	50	AES/DES standard	4.8	0.006
Data B	46.3	50	AES/DES standard	3.7	0.005
Data C	44.9	50	AES/DES standard	5.1	0.006
Data D	45.7	50	AES/DES standard	4.3	0.006
Average	45.53	50	/	4.47	0.0058

Table 3 shows the avalanche effect and cryptographic attack resistance indicators of the proposed DOKS encryption algorithm on four types of datasets. Firstly, from the perspective of average bit flip rate, when there is a 1-bit perturbation in plaintext, the average bit flip rate of ciphertext remains between 44.9% and 46.3%, with an overall average of 45.53%. Although slightly lower than the ideal benchmark of 50%, the fluctuation range is controlled between 3.7% and 5.1%, indicating that the algorithm has approached the level of classical block ciphers such as AES and DES in terms of diffusion performance. Secondly, the resistance to differential attacks is reflected by the DP bias value, which shows that the bias of each dataset is within 5%. This means that small plaintext differences can be fully amplified in ciphertext, thereby reducing the feasibility of attackers using differential statistical rules to derive keys. In terms of resisting linear cryptanalysis, the LP_max coefficient is always less than 0.006, with a value close to zero, indicating that there is no significant linear correlation between plaintext, ciphertext, and subkeys. The linear approximation probability of the algorithm is extremely low, making it difficult to be successfully modeled by linear approximation methods. In addition, the performance of the four sets of data is relatively balanced, and there is no abnormal low performance in any specific dataset, indicating that the algorithm maintains stable diffusion and resistance performance under different data distributions. Overall, the experimental results validate that the proposed encryption method can effectively resist both differential and linear typical cryptographic attacks while maintaining strong diffusion characteristics, providing a reliable security foundation for its practical application in cloud edge collaborative environments. The performance of the encryption model in different data is analyzed, as shown in Figure10.



(a) Bad pixel rate

(b) Safety

Figure 10: Analysis of bad pixel rate and security of SDE model

Figs.10 (a) and (b) show the trends in bad pixel rate and security for different datasets. In Figure10 (a), the bad pixel rate of Data C steadily increases with the increase of experimental times, but does not exceed the threshold of 3%. The bad pixel rates of Data A to D are relatively stable and far below 3%. Among them, the bad pixel rate of Data B and Data D is relatively low, maintained at around 1%, indicating that their performance in the experiment is more reliable and they have strong anti-interference ability. Although Data A initially has a slightly lower defect rate, as the number of experiments increases, its defect rate gradually increases, approaching 2%, possibly due to fluctuations in the quality of the dataset or external factors. In Figure10 (b), the security of Data D and Data B consistently remains at a high level, close to 1, indicating that these two datasets exhibit strong resistance to attacks and security during encryption or processing. The security of Data A and Data C has some fluctuations, especially Data C. Although the initial security is high, as the number of experiments increases, its security gradually decreases, possibly due to insufficient encryption mechanisms or algorithms in the processing of the dataset, resulting in a decrease in security. In summary, the proposed method can perform well on different data. Table 4 analyzes the performance of various SDE models.

Table 4: Performance analysis of SDE model

Metric	Data A	Data B	Data C	Data D
Accuracy	0.92	0.94	0.88	0.91
Precision	0.89	0.92	0.85	0.89
Recall	0.91	0.93	0.87	0.9
F1 value	0.90	0.92	0.86	0.89
AUC	0.95	0.97	0.93	0.96
Bad Pixel Rate (%)	1.5	1.2	2.5	1.0
Safety	0.87	0.90	0.83	0.88
Processing Time (s)	5.2	4.8	6.1	5.0

In Table 4, the model has the highest accuracy and precision in Data B, indicating good predictive performance and accurate identification of positive samples. In contrast, the accuracy (0.88) and precision (0.85) of Data C are lower, but still remain above 0.8. In terms of recall rate, Data B is also leading, indicating that it can effectively identify positive samples, while Data C's recall rate is 0.87, which is slightly insufficient. In terms of F1 value, Data B performs the best, reaching 0.92, with excellent overall precision and recall performance. The highest security score for Data B is 0.90, indicating that it performs well in terms of protection and encryption. In terms of processing time, Data B has the lowest 4.8 seconds, reflecting its high processing efficiency. This indicates that the proposed method can perform well on different data. To further assess the cryptographic strength of the proposed SDE algorithm, this section evaluates four key properties commonly used in encryption algorithm analysis: information entropy, key sensitivity, Number of Pixels Change Rate (NPCR), and Unified Average Changing Intensity (UACI). These metrics are calculated by applying the DOKS encryption module to sample textual data and analyzing its ciphertext under controlled perturbations. Table 5 shows the evaluation results based on 10 independent tests across sample sensitive data fields such as names, ID numbers, and email addresses.

Table 5: Cryptographic security metrics of the proposed SDE algorithm

Metric	Ideal Value	Observed Average	Standard Deviation
Information Entropy	8	7.95	±0.02
Key Sensitivity (%)	~100%	99.21	±0.34
NPCR (%)	≥ 99.00	99.06	±0.27
UACI (%)	≥ 33.00	33.41	±0.19
CCA Success Rate (%*)	→ 0	< 0.8	/
LP_max	→ 0	< 0.006	/

Overall, the entropy of ciphertext information reaches 7.95, which is close to the theoretically optimal value of 8, indicating that the encrypted output is close to uniform in statistical distribution, making it difficult for attackers to obtain useful information through frequency analysis. The key sensitivity experiment shows that when only one bit of the key is modified, the average difference rate of the ciphertext reaches 99.21%, which is close to the ideal value of 100%. This proves that DOKS can ensure that even minimal key perturbations can cause drastic changes in the ciphertext, effectively preventing key recovery attacks. NPCR and UACI are 99.06% and 33.41%, respectively, both of which meet the internationally recognized symmetric cryptographic security standards. This indicates that the encryption system has strong diffusion and equalization characteristics at the pixel or bit level, and can resist differential and correlation analysis. In the newly added resistance test to ciphertext attacks, the success rate of attackers attempting to infer sub keys through ciphertext perturbation was less than 0.8%, far below the common exploitable threshold, indicating that the algorithm exhibits high robustness in CCA scenarios. The maximum correlation coefficient LP_max of the linear approximation experiment is less than 0.006, which is close to zero, indicating that the linear correlation between the DOKS ciphertext output and plaintext input is extremely low, greatly reducing the feasibility of linear cryptanalysis. Based on various indicators, DOKS demonstrates empirical resistance to differential attacks, linear analysis, and selective ciphertext attacks while maintaining high entropy and strong diffusion, further verifying its feasibility and security foundation as a sensitive information encryption method for cloud edge collaborative environments. Table 6 presents a unified comparison across two stages of the proposed system pipeline: sensitive entity recognition, and data encryption.

Table 6: Comprehensive comparison of NER models and encryption algorithms

Method/Algorithm	Task Type	Accuracy/Enc. Time	F1/Dec. Time	Model Size/Key Type
BiLSTM-CRF	NER	0.99	0.94	1.5M params
BERT-base	NER	0.987	0.95	110M params
RoBERTa-base	NER	0.988	0.95	125M params
DOKS (proposed)	Encryption	134 ms	183 ms	Stream key
AES-128	Encryption	92 ms	84 ms	128-bit symmetric
RSA-2048	Encryption	1582 ms	2114 ms	2048-bit public key
Paillier (HE)	Encryption	5723 ms	6940 ms	Homomorphic public key

In the NER task, while transformer-based models such as BERT and RoBERTa achieve high F1 values (0.95), the BiLSTM-CRF model reaches a comparable 0.94 with significantly smaller model size and faster training convergence, making it ideal for edge deployment. In the encryption task, DOKS exhibits a strong trade-off between efficiency and adaptive content-level protection. Although AES is faster, it lacks semantic awareness. RSA and Paillier encryption provide higher security guarantees but are prohibitively slow for real-time text processing. These results demonstrate that the proposed BiLSTM-CRF+DOKS architecture strikes a balanced performance profile for cloud-edge collaborative environments.

To conduct theoretical cryptographic verification, based on the construction of a symmetric encrypted communication model, this paper introduces SHA-256 hash function and 128-bit random salt to generate initial keys based on the DOKS key stream generation mechanism, and combines the dynamic diffusion process of pseudo-random number streams to construct an encryption structure that satisfies the IND-CPA model. During the experiment, a systematic evaluation was conducted on the dimensions of plaintext perturbation, key perturbation, and ciphertext statistics. The results showed that the DOKS algorithm achieved an average change rate of 45.53% in the encryption results when the key perturbation was 1 bit in the avalanche effect test, which meets the high diffusion requirements; The information entropy index reaches 7.95, close to the ideal maximum value of 8.0, indicating a highly random distribution of ciphertext; The NPCR (ciphertext pixel change rate) reaches 99.34%, and the UACI (average brightness change) is 33.26%, both of which meet the security standards commonly used in the field of image encryption. In addition, from a theoretical analysis perspective, DOKS uses SHA-256 and 128 bit random salt to generate 256 bit session keys, combined with pseudo-random stream generator and dynamic replacement mechanism, which can meet the IND-CPA security model and resist plaintext selection attacks while ensuring the uniqueness of each keystream round; When the Encrypt then MAC structure is deployed in conjunction with message authentication mechanisms such as HMAC, it can be further extended to IND-CCA security, and the encryption scheme will have the ability to resist ciphertext selection attacks. In theoretical derivation, the probability of salt collision is approximately 2.7 × 10^{-20} at 2^{32} sessions, which is far below the acceptable threshold, ensuring the robustness of the key update mechanism. Overall, DOKS not only has strong statistical obfuscation diffusion capability in terms of security, but also meets the requirements of formal cryptographic security models.

In addition to avalanche effect and entropy tests, the proposed DOKS algorithm was analyzed against common cryptographic attack strategies. For brute-force attacks, the use of a SHA-256 based key derivation with a 128-bit random salt results in an effective key space of 2^{256}, which makes exhaustive key search computationally infeasible with current technology. For differential cryptanalysis, the observed NPCR of 99.34% and UACI of 33.26% indicate that a single-bit change in the plaintext or key produces highly uncorrelated ciphertext outputs, making it difficult for an adversary to exploit differential patterns. For linear cryptanalysis, the non-linear substitutions and matrix transformations within the Feistel rounds ensure that no simple linear approximations can predict ciphertext behavior; the measured linear probability (LPmax) is approximately 2^{-128}, which is negligible. Taken together, these results demonstrate that the DOKS encryption scheme exhibits strong resistance to brute-force, differential, and linear attacks, complementing the empirical metrics of avalanche effect and entropy already reported.

5 Conclusion

Aiming at the challenge of data encryption in cloud computing and edge computing environments, this research proposed an efficient SDE algorithm based on AI. This study combined DOKS and BiLSTM-CRF models to achieve efficient extraction and encryption of sensitive information. In the experiment, the BiLSTM-CRF model performed well in key performance indicators such as accuracy, recall, and F1 value, with an accuracy of 0.99 and an F1 value of 0.94, superior to the LSTM-CRF and LSTM models. Further performance analysis showed that BiLSTM-CRF outperformed other models in terms of encryption and decryption speed. When processing 100KB of data, the encryption time was 134 milliseconds and the decryption time was 183 milliseconds, significantly lower than LSTM and LSTM-CRF. During the training process, BiLSTM-CRF had a faster convergence speed, achieving 90% accuracy in just 28 rounds, while LSTM and LSTM-CRF required 41 and 49 rounds. In terms of security, during testing, this method showed strong resistance to attacks, with security consistently maintained above 0.90 on the Data B and Data D datasets. Comparing the encryption performance of different datasets, Data B performed the best with an accuracy of 0.94, security of 0.90, and processing time of 4.8 seconds, demonstrating efficient and reliable performance. However, there are still certain shortcomings in this study, especially in terms of adaptability to different datasets and stability of encryption models, with some datasets having lower processing efficiency. Future research can further explore how to improve performance in large-scale data and complex scenarios by optimizing model structures, improving encryption algorithms, and enhancing algorithm adaptability in different environments.

Although the DOKS algorithm has shown good empirical results in terms of obfuscation strength and performance efficiency, it has not yet provided a comprehensive theoretical security analysis, such as resistance to differential, linear, or selective ciphertext attacks. Future research will focus on formal cryptographic proofs and adversarial model evaluations to rigorously validate the security guarantees of DOKS under practical deployment conditions. Although this study focuses primarily on algorithm design and performance evaluation, the proposed BiLSTM-CRF+DOKS framework is compatible with real-world deployment in cloud-edge architectures. The BiLSTM-CRF model, due to its low parameter size (1.5M), can be deployed on edge devices with limited computational capacity, such as ARM-based chips or edge gateways. Encryption modules based on DOKS are stream-oriented and stateless, allowing flexible embedding into microcontroller-based IoT nodes. Key generation based on user-specific features and a deterministic salt can support decentralized key management without heavy infrastructure. Future work will consider integrating this model with existing IoT security frameworks such as MQTT-TLS, OPC-UA, or AWS IoT Core, and optimizing runtime adaptability under edge resource fluctuations.

Despite demonstrating strong performance across multiple evaluation criteria, the proposed framework still has certain limitations. First, the avalanche effect value measured at 45.53 percent, although acceptable, deviates from the ideal theoretical benchmark of 50 percent, which implies that the diffusion capability is not fully maximized. Second, the current evaluation relies on a single dataset (MSRA NER) focused on Chinese entity recognition, and the absence of multilingual or cross-domain datasets constrains the assessment of generalizability. Third, the selective encryption strategy, while efficient, inherently leaks the positions and lengths of non-encrypted fields, which could be exploited in adversarial scenarios. Finally, theoretical security proofs have been discussed but not exhaustively formalized within simulation-based cryptographic models, leaving a gap between empirical validation and provable security. These constraints highlight avenues for further refinement and broader validation of the proposed system.

Fundings

The research is supported by: The “Qinglan Project” of Golleges and Universities in Jiangsu Province, “Sudian Yingcai” Project, 2023Basic Science (Natural Science) research project of Jiangsu Colleges and universities: Research and design of low-frequency, low-noise, wide-dynamic range current readout circuit(23KJD510001),the Natural Science Foundation of Jiangsu Vocational College of Electronics and Information(JSEIY2023006), and 2024Basic Science(Natural Science) research project of Jiangsu Colleges and universities: Research on Knowledge Discovery and Feature Selection Methods for Hybrid Data (24KJD520002).

References

[1] Kaur, G., Agarwal, R., & Patidar, V. (2022). Color image encryption scheme based on fractional Hartley transform and chaotic substitution-permutation. The Visual Computer, 38(3), 1027-1050. https://doi.org/10.1007/s00371-021-02066-w

[2] Senthilkumar M, Murugan BS. (2022). Enhancing The Security of An Organization from Shadow Iot Devices Using Blow-Fish Encryption Standard. Acta Informatica Malaysia, 6(1): 22-24. http//:doi.org/ 10.26480/aim.01.2022.22.24

[3] Aldabbas, H. (2023). Efficient bandwidth allocation in SDN-based peer-to-peer data streaming using machine learning algorithm. Journal of Supercomputing, 79(6), 6802-6824. https://doi.org/10.1007/s11227-022-04929-y

[4] Sun, W. L., Tang, Y. H., & Huang, Y. L. (2023). HiRAM: A hierarchical risk assessment model and its implementation for an industrial Internet of Things in the cloud. Software Testing, Verification and Reliability, 33(5), 1-27. https://doi.org/10.1002/stvr.1847

[5] Liu, D., Zhen, H., Kong, D., Chen, X., Zhang, L., Yuan, M., & Wang, H. (2021). Sensor’s anomaly detection of industrial Internet of Things based on isolated forest algorithm and data compression. Scientific Programming, 21(1), 254-261. https://doi.org/10.1155/2021/6699313

[6] Kumar, R., & Tripathi, R. (2021). DBTP2SF: A deep blockchain-based trustworthy privacy-preserving secured framework in industrial internet of things systems. Transactions on Emerging Telecommunications Technologies, 32(4), 25-37. https://doi.org/10.1002/ett.4222

[7] Wu, H., Tian, H., Fan, S., & Ren, J. (2020). Data age aware scheduling for wireless powered mobile-edge computing in industrial Internet of Things. IEEE Transactions on Industrial Informatics, 32(12), 12-32.

[8] Yang, L., Zou, W., Wang, J., & Tang, Z. (2022). Edge Share: A blockchain-based edge data-sharing framework for Industrial Internet of Things. Neurocomputing, 485(7), 219-232. https://doi.org/10.1016/j.neucom.2021.01.147

[9] Gilmolk, A. M. N., & Aref, M. R. (2024). Lightweight image encryption using a novel chaotic technique for the safe Internet of Things. International Journal of Computational Intelligence Systems, 17(1), 146-152. https://doi.org/10.1007/s44196-024-00535-3

[10] Peng, X., & Zeng, Y. (2020). Image encryption application in a system for compounding self-excited and hidden attractors. Chaos, Solitons & Fractals, 139(6), 1144-1159. https://doi.org/10.1016/j.chaos.2020.110044

[11] Boussif, M., Aloui, N., & Cherif, A. (2020). Securing DICOM images by a new encryption algorithm using Arnold transform and Vigenère cipher. IET Image Processing, 14(6), 1209-1216. https://doi.org/10.1049/iet-ipr.2019.0042

[12] Wang, X., & Gao, S. (2020). Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network. Information Sciences, 539(9), 195-214. https://doi.org/10.1016/j.ins.2020.06.030

[13] Jain, K., Aji, A., & Krishnan, P. (2021). Medical image encryption scheme using multiple chaotic maps. Pattern Recognition Letters, 152(12), 356-364. https://doi.org/10.1016/j.patrec.2021.10.033

[14] Zarebnia, M., Pakmanesh, H., & Parvaz, R. (2019). A fast multiple-image encryption algorithm based on hybrid chaotic systems for gray scale images. Optik, 179(3), 761-773. https://doi.org/10.1016/j.ijleo.2018.10.025

[15] Hu, W., & Dong, Y. (2022). Quantum color image encryption based on a novel 3D chaotic system. Journal of Applied Physics, 131(11), 1142-1155. https://doi.org/10.1063/5.0084611

[16] Huang, H., Yang, S., & Ye, R. (2020). An efficient symmetric image encryption by using a novel 2D chaotic system. IET Image Processing, 14(6), 1157-1163. https://doi.org/10.1049/iet-ipr.2019.0551

[17] Nie, Z., Liu, Z. X., He, X. T., & Gong, L. H. (2019). Image compression and encryption algorithm based on advanced encryption standard and hyper-chaotic system. Optica Applicata, 49(4), 545-558. https://doi.org/10.37190/oa190402

[18] Li, B., Feng, Y., Xiong, Z., Yang, W., & Liu, G. (2021). Research on AI security enhanced encryption algorithm of autonomous IoT systems. Information Sciences, 575(3), 379-398. https://doi.org/10.1016/j.ins.2021.06.016

[19] Yao, M., Chen, Z., Deng, H., et al. (2025). A color image compression and encryption algorithm combining compressed sensing, Sudoku matrix, and hyperchaotic map. Nonlinear Dynamics, 113(3), 2831-2865. https://doi.org/10.1007/s11071-024-10334-2

[20] Singh, D., & Kumar, S. (2025). Image authentication and encryption algorithm based on RSA cryptosystem and chaotic maps. Expert Systems with Applications, 274(15), 141-152. https://doi.org/10.1016/j.eswa.2025.126883

[21] Wang, A., Shen, C., Pan, J., et al. (2023). Research on multiple-image encryption method using modified

Gerchberg-Saxton algorithm and chaotic systems. Optical Engineering, 62(9), 098103. https://doi.org/10.1117/1.OE.62.9.098103

[22] Liu, Y., & Xue, R. (2024). 3D medical image encryption algorithm using biometric key and cubic S-box. Physica Scripta, 99(5), 55035-55055. https://doi.org/10.1088/1402-4896/ad3b3d

[23] Tong, X., Liu, X., & Pan, T. (2024). A visually meaningful secure image encryption algorithm based on conservative hyperchaotic system and optimized compressed sensing. Multimedia Systems, 30(3), 168-172. https://doi.org/10.1007/s00530-024-01370-4

[24] Alawida, M. (2024). A novel image encryption algorithm based on cyclic chaotic map in industrial IoT environments. IEEE Transactions on Industrial Informatics, 20(8), 10530-10541. https://doi.org/10.1109/TII.2024.3395631

[25] Erkan, U., Toktas, A., Toktas, F., & Alenezi, F. (2022). 2D eπ-map for image encryption. Information Sciences, 589, 770-789. https://doi.org/10.1016/j.ins.2021.12.126

[26] Jin, B., Fan, L., & Zhang, B. (2024). Image encryption hiding algorithm based on digital time-varying delay chaos model and compression sensing technique. Microelectronics Journal, 27(9), 141-153. https://doi.org/10.1016/j.isci.2024.110717

[27] Singh, P., Acharya, B., & Chaurasiya, R. K. (2022). Low-area and high-speed hardware architectures of Lblock cipher for Internet of Things image encryption. Journal of Electronic Imaging, 31(3), 033012. https://doi.org/10.1117/1.JEI.31.3.033012

[28] Chen, L. W., Tsai, K. L., & Leu, F. Y. (2024). Time parameter based low-energy data encryption method for mobile applications. *CMES-Computer Modeling in Engineering & Sciences, 140*(3), 2779-2794. https://doi.org/10.32604/cmes.2024.052124

[29] Dzemyda, G., Sabaliauskas, M., & Medvedev, V. (2022). Geometric MDS performance for large data dimensionality reduction and visualization. Informatica, 33(2), 299-320. https://doi.org/10.15388/22-INFOR491

[30] Mehta, P., Aggarwal, S., & Tandon, A. (2023). The effect of topic modelling on prediction of criticality levels of software vulnerabilities. Informatica, 47(6), 283-304. https://doi.org/10.31449/inf.v47i6.3712