DOI: https://doi.org/10.53052/9788367652124.18

Kuanysh SYZDYKBAYEV[1]

Opiekun naukowy: Aigul SHAIKHANOVA[2], Vasyl MARTSENYUK[3]

# BADANIA I OPRACOWANIE ALGORYTMÓW SZYFROWANIA INFORMACJI TEKSTOWYCH W OPARCIU O SEKWENCJE CHAOTYCZNE

**Streszczenie:** W artykule zbadano synergię map chaotycznych i uczenia maszynowego w szyfrowaniu tekstu, przedstawiając algorytm wykorzystujący dynamikę chaotyczną i uczenie maszynowe w celu zwiększenia bezpieczeństwa. Przegląda chaotyczne mapy, AES, Blowfish i aplikacje do uczenia maszynowego, podkreślając ich rolę w udoskonalaniu metod szyfrowania tekstu.

**Słowa kluczowe:** mapy chaotyczne, uczenie maszynowe, szyfrowanie tekstu, kryptografia

# RESEARCH AND DEVELOPMENT OF TEXT INFORMATION ENCRYPTION ALGORITHMS BASED ON CHAOTIC SEQUENCES

**Summary:** The paper investigates the synergy of chaotic maps and machine learning for text encryption, presenting an algorithm leveraging chaotic dynamics and machine learning to enhance security. It reviews chaotic maps, AES, Blowfish, and machine learning applications, emphasizing their roles in advancing text encryption methods.

**Keywords:** chaotic maps, machine learning, text encryption, cryptography

## 1. Introduction

In recent years, the development of various cryptographic techniques has become crucial to protect sensitive information, particularly medical data. Among these techniques, the combination of chaotic maps and machine learning networks has shown promising results. One study conducted by Lin et al. proposes a symmetric cryptography technique that uses a chaotic map and multilayer machine learning

[1] Post-graduate, L.N.Gumilyov Eurasian National University, email: kukapvl@mail.ru
[2] PhD, Professor of Department of Information Security, L.N.Gumilyov Eurasian National University, email: aigul.shaikhanova@gmail.com
[3] Prof. Dr hab., University of Bielsko-Biala, Faculty of Mechanical Engineering and Computer Science, email: vmartsenyuk@ubb.edu.pl

network to secure physiological signal infosecurity, specifically electrocardiogram data [1]. Similarly, Rupa et al. also explore the use of a deep learning-based chaotic logistic map to secure multimedia data [2]. On the other hand, Liu et al. and Pan et al. apply machine learning algorithms to intelligent encryption for digital information of real-time image text and financial security system, respectively [3-5]. Zhou et al. introduce a novel image encryption cryptosystem based on true random numbers and chaotic systems [4]. Another recent work by He et al. proposes a new image encryption algorithm that utilizes the Optimal Filter Long Short-Term Memory (OF-LSTMs) and chaotic sequences to achieve high levels of security. The authors' experiments show that their proposed algorithm is robust against various attacks and outperforms several state-of-the-art image encryption methods [6]. Lastly, Ding et al. present DeepKeyGen, a deep learning-based stream cipher generator for medical image encryption and decryption. The authors use a convolutional neural network (CNN) to learn the characteristics of medical images and generate a stream cipher key. Their experiments demonstrate that DeepKeyGen provides strong security and efficiency for medical image encryption and decryption compared to other state-of-the-art methods in terms of security and efficiency [7]. Yan et al. also contribute to the field by presenting a chaotic image encryption algorithm based on arithmetic sequence scrambling model and DNA encoding operation [8]. These studies demonstrate the significant potential of combining chaotic maps and machine learning techniques for secure communication and data protection.

There have been various recent developments in the field of data security using chaotic systems, machine learning, and other techniques. These methods have shown promising results in terms of security and efficiency, and they are expected to play an increasingly important role in the future of data security.

Image encryption has been extensively researched, while text encryption using chaotic sequences and machine learning has received relatively little attention. This method of encryption holds great potential in ensuring the security of sensitive data, especially in the context of text-based data.

## 2. Chaotic maps used for encryption

There are several chaotic maps utilized for encryption, and these are the primary ones:

### 2.1. The logistic map

The logistic map is a one-dimensional nonlinear map that exhibits chaotic behavior under certain conditions. It is defined by the equation:

$$x[n+1] = r \cdot x[n] \cdot (1 - x[n]) \tag{1}$$

where *x[n]* is the value of the variable at time *n*, and *r* is a parameter that controls the rate of divergence. The map is typically visualized by plotting the value of *x[n]* versus *n*.

The logistic map exhibits a wide range of behaviors depending on the value of the parameter r. For values of r less than 1, the map converges to a stable fixed point. As

*r* increases, the map undergoes a period-doubling bifurcation, producing a periodic orbit with a period of 2. This process repeats as r increases further, leading to more complex periodic orbits with periods of 4, 8, 16, and so on.
At a critical value of *r*, the map undergoes a transition to chaotic behavior, characterized by sensitive dependence on initial conditions and aperiodic orbits. The chaotic behavior of the logistic map has been studied extensively and has applications in fields such as physics, biology, and cryptography.
The logistic map has also been used in various encryption algorithms, as it can be used to generate a pseudorandom sequence of numbers that can be used to scramble data. However, its use in encryption has been criticized due to the fact that the map is deterministic and can potentially be hacked through various attacks [9].

### 2.2. The Henon map

The Henon map is a two-dimensional nonlinear map that exhibits chaotic behavior. It is defined by the equations:

$$\begin{aligned} x[n+1] &= 1 - a \cdot x[n]^2 + y[n] \\ y[n+1] &= b \cdot x[n] \end{aligned} \tag{2}$$

where *x[n]* and *y[n]* are the values of the variables at time *n*, and *a* and *b* are parameters that control the behavior of the map. The map is typically visualized by plotting the value of *y[n]* versus *x[n]*.
The Henon map exhibits a wide range of behaviors depending on the values of the parameters *a* and *b*. At certain parameter values, the map exhibits chaotic behavior, characterized by sensitive dependence on initial conditions and aperiodic orbits. The chaotic behavior of the Henon map has been studied extensively and has applications in fields such as physics, biology, and cryptography.
The Henon map has also been used in various encryption algorithms, as it can be used to generate a pseudorandom sequence of numbers that can be used to scramble data. However, like other chaotic maps, the Henon map is deterministic and can potentially be hacked through various attacks. Therefore, researchers continue to explore the potential uses of chaotic maps in encryption and other fields of study [10].

### 2.3. The Lorenz system

The Lorenz system is a set of three coupled nonlinear differential equations that exhibit chaotic behavior. It was introduced by Edward Lorenz in 1963 as a simplified model of atmospheric convection, and has since been studied extensively in the fields of physics, engineering, and mathematics.
The Lorenz system is defined by the equations:

$$\frac{dx}{dt} = \sigma(y - x) \qquad \frac{dy}{dt} = x(\rho - x) \qquad \frac{dz}{dt} = xy - \beta z \tag{3}$$

where *x, y*, and *z* are the variables that describe the state of the system, and $\sigma$, $\rho$, and $\beta$ are parameters that control the behavior of the system. The system exhibits chaotic behavior for certain values of the parameters, characterized by sensitive dependence on initial conditions and aperiodic orbits.

The Lorenz system has been used in various applications, including weather forecasting, fluid dynamics, and cryptography. However, its use in encryption has been criticized due to the fact that the system is deterministic and can potentially be hacked through various attacks [11].
The Lorenz system, Henon map, and Logistic map have different advantages and limitations in terms of implementation results in cryptography. The Lorenz system has good security properties but requires high computational resources, while the Henon map and Logistic map have good performance but may be vulnerable to some attacks. The choice of which system to use in cryptography depends on the specific requirements of the application and the desired security properties [12].

## 3. AES and Blowfish encryption algorithm

AES (Advanced Encryption Standard) is commonly used in text encryption to protect the confidentiality of messages or data being transmitted or stored. In text encryption, plaintext is transformed into ciphertext using AES, which makes it unreadable to anyone who does not have the key to decrypt it.
AES can be used in a variety of text encryption applications, including email, instant messaging, file sharing, and cloud storage. For example, AES can be used to encrypt email messages to prevent unauthorized access to the content of the email. AES can also be used to encrypt files stored in the cloud to prevent unauthorized access to the data.
To use AES for text encryption, a key is needed to encrypt and decrypt the data. The same key is used for both encryption and decryption, so it must be kept secret to maintain the security of the encrypted data. The key can be generated by a random number generator or derived from a passphrase using a key derivation function.
AES supports three different key sizes: 128-bit, 192-bit, and 256-bit. The larger the key size, the stronger the encryption, but also the slower the encryption and decryption process. The choice of key size depends on the specific security requirements of the application.
Blowfish is a symmetric-key encryption algorithm that can be used for text encryption. It was designed by Bruce Schneier in 1993 as a fast, free alternative to existing encryption algorithms.
Blowfish uses a variable-length key, which means that the key size can range from 32 bits to 448 bits. The algorithm consists of 16 rounds of a Feistel network, which is a type of encryption algorithm that uses multiple rounds of substitution and permutation to produce ciphertext.
In text encryption, Blowfish can be used to encrypt messages, files, and other types of data. To use Blowfish for text encryption, a key is needed to encrypt and decrypt the data. The same key is used for both encryption and decryption, so it must be kept secret to maintain the security of the encrypted data. The key can be generated by a random number generator or derived from a passphrase using a key derivation function.
Both AES and Blowfish are widely used encryption algorithms that can provide strong protection for text data. However, there are some differences between them that may make one better suited for a particular application than the other.

AES is considered to be one of the most secure encryption algorithms available and is widely used in many applications, including text encryption. It has been extensively analyzed and tested by the cryptographic community and is considered to be highly resistant to attacks.
Blowfish is also a widely used encryption algorithm, but it is not as commonly used as AES. It is a symmetric-key algorithm that uses a variable-length key, which means that the key size can range from 32 bits to 448 bits. Blowfish is generally considered to be a fast encryption algorithm, making it suitable for applications where speed is important.
In terms of security, both AES and Blowfish are considered to be secure encryption algorithms, but AES is generally considered to be more secure due to its larger key sizes and stronger cryptographic properties. Additionally, AES is the current standard for encryption and is widely used in many applications, including text encryption.
Overall, while both AES and Blowfish can provide strong protection for text data, AES is generally considered to be the better choice for text encryption due to its stronger security properties and widespread use [13].

## 4. Machine learning

There are different ways in which machine learning can be used for text encryption. One approach is to use machine learning algorithms to create a model that can automatically generate strong encryption keys or password for the encryption process. Another approach is to use machine learning algorithms to create more complex encryption algorithms that are harder to crack.
For example, machine learning algorithms such as neural networks can be used to generate strong encryption keys based on patterns in input data. These algorithms can analyze large amounts of data to find patterns and generate unique keys that are difficult to guess. In addition, machine learning algorithms can be used to create more complex encryption algorithms based on deep learning techniques, which can be harder to crack by attackers.
Another way machine learning can be used in text encryption is through the detection of malicious or fraudulent behavior. For example, machine learning algorithms can be trained to detect patterns in encrypted data that are indicative of malicious activity, such as an attacker attempting to decrypt the data. This can help to prevent cyber attacks and protect sensitive data from being compromised.
Machine learning can be used to analyze encrypted text by using various techniques such as homomorphic encryption, secure multiparty computation, and differential privacy.
Homomorphic encryption allows computation on encrypted data without the need to decrypt it. This technique is useful for privacy-preserving machine learning applications, where the data owner encrypts their data and sends it to a model owner who trains a model on the encrypted data without ever seeing the raw data. This approach allows the data owner to keep their data private while still benefiting from the model's predictions.
Secure multiparty computation (SMC) is another technique used for analyzing encrypted text. It allows multiple parties to compute a function on their inputs without revealing their private data to each other. SMC can be used for privacy-preserving

machine learning applications, where multiple parties contribute their encrypted data to train a model without revealing their data to each other.
Differential privacy is a technique used to protect individual privacy while still allowing analysis of data. It adds noise to the data to ensure that individual records cannot be identified, while still providing meaningful results. Differential privacy can be used for machine learning applications where the goal is to protect the privacy of the individuals whose data is being used to train the model [14].

## 5. Algorithm for text encryption based on random sequence

Using the methods and techniques presented above, we can formulate the general approach as follows. Namely, firstly we generate a random sequence using a chaotic dynamic system of prey-predator type with discrete delay. Delay plays the role of a bifurcation parameter leading to chaotic behavior. The initial conditions and threshold value play the role of the secret key. Firstly, we generate the chaotic trajectory of the model flowing from the initial conditions. Further, the threshold value is used in the following way. We construct the sequence of the markers by replacing the corresponding element of the sequence with 1 if the element is greater than the threshold, and 0 otherwise. Secondly, we apply the XOR operation of chars with the elements of the chaotic trajectory marked by ones. For better encryption results the value of the threshold can be chosen with the help of machine learning.
So, the algorithm consists of several steps:

**Algorithm: Chaotic Sequence Text Encryption**
Input:

- **initial_condition**: Initial condition for the chaotic system.
- **threshold**: Threshold value for constructing markers.
- **num_elements**: Number of elements in the text.
- **r**: Parameters for the chaotic dynamic system.
- $\boldsymbol{\tau}$: Discrete delay for chaotic sequence generation.
- **text**: The text to be encrypted.

Output:

- **encrypted_text**: The encrypted text.

Steps:

1. **Generate Chaotic Sequence:**
    - Initialize **x** with the **initial_condition**.
    - For each iteration from 1 to **num_elements + delay**:
        - Update **x** using the dynamic map: $x_{\{n+1\}} = f(x_n, r, \tau)$.
        - Append the updated **x** to the chaotic sequence.
2. **Create Markers:**
    - Generate a sequence of markers by comparing each element of the chaotic sequence with the **threshold**.
        - If the element is greater than the **threshold**, set the marker to 1.
        - Otherwise, set the marker to 0.

3. **Encrypt Text:**
    - For each character in the input **text**:
        - XOR the character with the corresponding marker from the generated sequence.
        - Append the result to the encrypted text.
4. **Output Result:**
    - Return the **encrypted_text**.

Example Usage:

```
initial_condition = 0.4
threshold = 0.6
num_elements = 1000
r = 3.8
delay = 5
text = "Hello, World!"
encrypted_text                                                    =
ChaoticSequenceEncryption(initial_condition,      threshold,
num_elements, r, delay, text)
```

This algorithm utilizes a chaotic sequence to generate markers based on a threshold, which are then used to XOR the characters of the input text for encryption. The chaotic dynamics, initial conditions, and threshold serve as the encryption key, providing a level of security to the text.

Below is a simple Python implementation using the map as a chaotic system.
import numpy as np

```
def f(x, r, delay):
    return r * x * (1 - x) # example of logistic map

def  generate_chaotic_sequence(initial_condition,  threshold,
num_elements, r, delay):
    chaotic_sequence = []
    x = initial_condition

    for _ in range(num_elements + delay):
        x = f(x, r, delay)
        chaotic_sequence.append(x)

    markers = [1 if element > threshold else 0 for element in
chaotic_sequence[delay:]]

    return markers

def encrypt_text(text, markers):
    encrypted_text = ''
    for i in range(len(text)):
        encrypted_char  =  chr(ord(text[i])  ^  markers[i  %
len(markers)])
        encrypted_text += encrypted_char
    return encrypted_text
```

```
def decrypt_text(encrypted_text, markers):
    return  encrypt_text(encrypted_text,  markers)    #  XOR
operation is its own inverse
```

This code defines functions for generating a chaotic sequence using the logistic map, creating markers based on a threshold, and encrypting and decrypting a given text using XOR with the markers. The chaotic sequence is used to create markers that are XORed with the text characters for encryption and decryption.

```
# Example usage:
initial_condition = 0.4
threshold = 0.6
num_elements = 1000
r = 3.8
delay = 5
chaotic_markers                                              =
generate_chaotic_sequence(initial_condition,      threshold,
num_elements, r, delay)

# Text to be encrypted
original_text = "Hello, World!"

# Encrypt the text
encrypted_text = encrypt_text(original_text, chaotic_markers)
print("Encrypted text:", encrypted_text)

# Decrypt the text
decrypted_text         =         decrypt_text(encrypted_text,
chaotic_markers)
print("Decrypted text:", decrypted_text)
```

## 6. Conclusions

In recent years, the fusion of chaotic maps and machine learning techniques has emerged as a promising avenue for enhancing cryptographic methods, particularly in securing sensitive information such as medical data. The surveyed literature showcases diverse applications, from securing physiological signal data like electrocardiograms to protecting multimedia data through deep learning-based chaotic logistic maps.

Several studies have demonstrated the synergy between chaotic systems and machine learning in diverse domains. Notable examples include the use of deep learning-based algorithms for intelligent encryption in real-time image text and financial security systems. Additionally, novel approaches, like the integration of true random numbers and chaotic systems in image encryption cryptosystems, highlight the continuous exploration of innovative techniques in the field.

The research landscape encompasses the development of image encryption algorithms using Optimal Filter Long Short-Term Memory (OF-LSTMs) and chaotic sequences,

proving robustness against various attacks and outperforming existing methods. Furthermore, the introduction of DeepKeyGen, a deep learning-based stream cipher generator for medical image encryption and decryption, exemplifies the potential of machine learning in generating secure cryptographic keys.

While image encryption has been extensively explored, text encryption using chaotic sequences and machine learning remains relatively unexplored. The proposed algorithm addresses this gap, offering a method to generate a random sequence using a chaotic dynamic system with discrete delay for text encryption. The integration of machine learning in selecting the threshold value enhances the security of the encryption process.

The paper also delves into the characteristics and applications of chaotic maps such as the logistic map, Henon map, and Lorenz system. These maps, despite having their advantages and limitations, contribute differently to the implementation results in cryptography. The selection of a particular chaotic system depends on the specific requirements of the application and the desired security properties.

Furthermore, the paper introduces the widely adopted Advanced Encryption Standard (AES) and Blowfish encryption algorithms for text encryption. While both algorithms offer strong protection for text data, AES is generally considered superior due to its larger key sizes and stronger cryptographic properties. The discussion underscores the importance of choosing encryption algorithms based on specific application requirements.

In the realm of machine learning, the paper explores various applications, including the generation of strong encryption keys, the development of complex encryption algorithms, and the detection of malicious behavior in encrypted data. The utilization of techniques like homomorphic encryption, secure multiparty computation, and differential privacy showcases the versatility of machine learning in enhancing text encryption while preserving privacy.

In summary, the paper highlights the evolving landscape of cryptographic techniques, emphasizing the symbiosis of chaotic maps and machine learning in securing diverse forms of sensitive information. The proposed algorithm for text encryption adds a novel dimension to this landscape, offering a potential avenue for ensuring the security of text-based data in the face of evolving threats.

## REFERENCES

1. LIN C. H., WU J. X., CHEN P. Y., LI C. M., PAI N. S., KUO C. L.: Symmetric Cryptography With a Chaotic Map and a Multilayer Machine Learning Network for Physiological Signal Infosecurity: Case Study in Electrocardiogram, in IEEE Access, vol. 9, pp. 26451-26467, 2021, doi: 10.1109/ACCESS.2021.3057586.
2. RUPA C., HARSHITHA M., SRIVASTAVA G., GADEKALLU T. R., MADDIKUNTA P. K. R.: Securing Multimedia Using a Deep Learning Based Chaotic Logistic Map, in IEEE Journal of Biomedical and Health Informatics, vol. 27, no. 3, pp. 1154-1162, March 2023, doi: 10.1109/JBHI.2022.3178629.

3. LIU L., GAO M., ZHANG Y. et al.: Application of machine learning in intelligent encryption for digital information of real-time image text under big data. J Wireless Com Network 2022, 21 (2022). https://doi.org/10.1186/s13638-022-02111-9.
4. ZHOU S., WANG X., ZHANG Y. et al.: A novel image encryption cryptosystem based on true random numbers and chaotic systems. Multimedia Systems 28, 95–112 (2022). https://doi.org/10.1007/s00530-021-00803-8
5. PAN S., WEI J.: HU S.: A Novel Image Encryption Algorithm Based on Hybrid Chaotic Mapping and Intelligent Learning in Financial Security System. Multimed Tools Appl 79, 9163–9176 (2020). https://doi.org/10.1007/s11042-018-7144-5.
6. HE Y., ZHANG YQ., HE X. et al.: A new image encryption algorithm based on the OF-LSTMS and chaotic sequences. Sci Rep 11, 6398 (2021). https://doi.org/10.1038/s41598-021-85377.
7. DING Y., TAN F., QIN Z., CAO M., CHOO K. K. R., QIN Z.: DeepKeyGen: A Deep Learning-Based Stream Cipher Generator for Medical Image Encryption and Decryption, in IEEE Transactions on Neural Networks and Learning Systems, 33(2022)9, 4915-4929, Sept. 2022, doi: 10.1109/TNNLS.2021.3062754.
8. YAN X., WANG X., XIAN Y.: Chaotic image encryption algorithm based on arithmetic sequence scrambling model and DNA encoding operation. Multimed Tools Appl 80(2021) 10949–10983, https://doi.org/10.1007/s11042-020-10218-8.
9. MURILLO-ESCOBAR M.A., CRUZ-HERNÁNDEZ C., CARDOZA-AVENDAÑO L. et al.: A novel pseudorandom number generator based on pseudorandomly enhanced logistic map. Nonlinear Dyn 87(2017), 407–425, https://doi.org/10.1007/s11071-016-3051-3
10. SUNEEL M.: Cryptographic pseudo-random sequences from the chaotic Hénon map. Sadhana 34, 689–701 (2009). https://doi.org/10.1007/s12046-009-0040-y.
11. ZUEV M. Y., LOGINOV S. S.: Practical Implementation of a Pseudo-Random Signal Generator Based on the Lorenz System Realized on FPGA, 2019 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), Russia, 2019, 1-4, doi: 10.1109/SYNCHROINFO.2019. 8814209.
12. SONG B., DING Q.: Comparisons of Typical Discrete Logistic Map and Henon Map. In: Pan, JS., Snasel, V., Corchado, E., Abraham, A., Wang, SL. (eds) Intelligent Data analysis and its Applications, Volume I. Advances in Intelligent Systems and Computing, 297(2014). Springer, Cham. https://doi.org/10.1007/978-3-319-07776-5_28.
13. RIZVI S. A. M., HUSSAIN S. Z., WADHWA N.: Performance Analysis of AES and TwoFish Encryption Schemes, 2011 International Conference on Communication Systems and Network Technologies, Katra, India, 2011, 76-79, doi: 10.1109/CSNT.2011.160.
14. PRADEEPTHI K. V., TIWARI V., SAXENA A.: Machine Learning Approach for Analysing Encrypted Data, 2018 Tenth International Conference on Advanced Computing (ICoAC), Chennai, India, 2018, pp. 70-73, doi: 10.1109/ICoAC44903.2018.8939101.