

Hybrid Neural Cryptography: A Novel Approach for Enhanced Security and Efficiency in Resource-Constrained Systems

Sourav Malakar

Assistant Professor, Department of CSE, Swami
Vivekananda University, Kolkata, WB, India

sourav.xaviers@gmail.com

Abstract

The advent of artificial intelligence and deep learning has revolutionized multiple fields, and cryptography is no exception. Traditional cryptographic algorithms, such as symmetric and asymmetric encryption, offer robust security but come with significant computational overhead, making them unsuitable for low-resource environments such as Internet of Things (IoT) devices and embedded systems. On the other hand, neural networks have demonstrated the ability to perform complex transformations that are difficult to reverse, providing potential solutions for cryptographic systems. In this paper, we propose a **Hybrid Neural Cryptography (HNC)** model that integrates classical cryptographic algorithms with artificial neural networks to improve both the security and computational efficiency of encrypted communication systems. We present the architecture of the HNC model, its cryptographic components, and evaluate its performance against traditional methods, demonstrating its effectiveness in enhancing security while maintaining low computational cost.

Introduction

The rise of IoT devices and mobile applications has prompted the need for cryptographic solutions that ensure data security without overwhelming the limited computational resources of these devices. Classical cryptographic algorithms, such as **AES (Advanced Encryption Standard)** for symmetric

encryption and **RSA (Rivest-Shamir-Adleman)** for asymmetric encryption, provide a high level of security but are computationally expensive, particularly for low-power devices. On the other hand, **neural cryptography**, which utilizes machine learning models (especially deep learning techniques like **Recurrent Neural Networks (RNNs)** or **Convolutional Neural Networks (CNNs)**), has shown promise for creating more efficient and adaptive cryptographic systems.

Neural cryptography has the potential to address some of the limitations of traditional cryptographic methods, but purely neural approaches often struggle with scalability, key management, and vulnerability to adversarial attacks. To overcome these challenges, we propose the **Hybrid Neural Cryptography (HNC)** model, which combines classical cryptography (AES, RSA) with neural network-based transformations. By doing so, we aim to combine the efficiency of classical algorithms with the advanced pattern recognition capabilities of neural networks, creating a cryptographic solution that is both secure and computationally feasible in low-resource environments.

Background and Motivation

Traditional Cryptographic Methods

Traditional cryptographic algorithms such as **AES** (symmetric) and **RSA** (asymmetric) have long been the standards for securing data. AES is widely used for encrypting bulk data, offering strong security with relatively fast encryption and decryption times, especially in modern processors. However, its efficiency diminishes in devices with limited processing capabilities, such as microcontrollers or IoT devices.

RSA, on the other hand, is primarily used for secure key exchange in communication systems. Its security is based on the difficulty of factoring large prime numbers, but RSA encryption and decryption are computationally intensive, especially with long key sizes (2048-bit or greater). This results in high computational costs, which is problematic for resource-constrained systems.

While these methods remain effective, their computational cost can be prohibitive in applications where low-power, low-latency performance is crucial.

Neural Cryptography and Machine Learning Approaches

Machine learning, and more specifically **deep learning**, has been used to design novel cryptographic schemes. Neural networks can learn complex, non-linear transformations, which may make it more difficult for an attacker to reverse engineer the encryption process. Research has shown that neural networks can be trained to generate cryptographic keys or perform encryption and decryption tasks directly.

However, neural cryptography is not without its challenges:

- **Scalability:** As the neural network model grows in size and complexity, it requires more computational resources and memory, making it unsuitable for low-resource devices.
- **Security Vulnerabilities:** Neural networks, particularly deep neural networks, are vulnerable to adversarial attacks. Small perturbations in input data can dramatically change the network's output, potentially breaking the encryption.

Thus, while neural networks provide some advantages in terms of security, their implementation as a sole cryptographic method has its limitations. A hybrid approach that leverages the strengths of both classical cryptography and neural networks offers an effective solution.

Hybrid Neural Cryptography (HNC) Model

The **Hybrid Neural Cryptography (HNC)** model is designed to combine the efficiency of classical cryptographic algorithms with the advanced capabilities of neural networks. It consists of the following key components:

Classical Cryptographic Layer

The classical cryptographic layer provides the base encryption and decryption functionality. This layer can use symmetric cryptography (e.g., **AES**) for data encryption and asymmetric

cryptography (e.g., **RSA**) for key exchange. The classical layer ensures that the fundamental security properties of the system—confidentiality, integrity, and authenticity—are met through well-established cryptographic techniques.

- **AES for Data Encryption:** AES is chosen for its computational efficiency, even in low-resource environments. In the HNC model, AES is used for the encryption of data with dynamically generated session keys.

- **RSA for Key Exchange:** RSA is used for securely exchanging symmetric encryption keys between the sender and the receiver. The use of asymmetric cryptography ensures that the session key can be exchanged securely over an open channel.

Neural Network Layer

The neural network layer is responsible for transforming the encrypted data generated by the classical cryptographic layer. This non-linear transformation adds an extra layer of complexity to the data, making it harder for an attacker to reverse engineer the encryption, even if they have access to the encrypted data.

- **Feedforward Neural Networks (FNN):** In our model, we use a **feedforward neural network (FNN)** to perform data transformations. The FNN can encode additional complexity by applying learned weights and biases to the encrypted data.

- **Convolutional Neural Networks (CNN):** For more complex transformations, we use **CNNs**, which are particularly effective in handling spatial data and applying convolutions to the encrypted data. This method helps further obscure the data from potential attackers.

The neural network processes encrypted data by encoding it into a seemingly random representation. The model is trained to generate these transformations based on an initial seed (e.g., a random key) provided by the classical cryptographic layer.

Key Management Layer

Key management is crucial in hybrid cryptographic systems, especially in neural cryptography. In the HNC model, keys are dynamically generated and managed by the neural network, which is trained to create secure session keys. This eliminates the need for static key management and reduces the risks associated with key distribution in traditional systems.

- **Neural Key Generation:** The neural network is trained to generate unique session keys that are used for AES encryption. These keys are dynamically created for each session and can be destroyed after use, enhancing security.

Security and Performance Evaluation

We evaluate the HNC model's performance on two major fronts: **security** and **computational efficiency**.

Security Evaluation

The security of the HNC model is assessed by examining its resilience to various cryptographic attacks:

- **Brute Force Attacks:** By adding a layer of neural network-based transformation, the HNC model makes brute-force attacks more difficult. An attacker would need not only the AES key but also knowledge of the neural network's weights and biases to successfully decrypt the data.

- **Adversarial Attacks:** Neural networks are typically susceptible to adversarial perturbations, where small changes to input data can result in large changes to the network's output. To mitigate this risk, the HNC model incorporates defensive measures, such as adversarial training, to make the network more robust against such attacks.

- **Known Plaintext Attacks:** The hybrid nature of the model ensures that even if part of the data is known, the neural network's transformation makes it difficult for attackers to reverse the encryption without access to the session key.

Performance Evaluation

Performance metrics such as **computation time**, **memory usage**, and **energy consumption** are analyzed in the context of low-resource systems (e.g., IoT devices, microcontrollers).

- **Computational Time:** Our tests demonstrate that while the HNC model introduces an additional processing step compared to traditional symmetric encryption alone, the overhead is minimal and well within acceptable limits for real-time applications.

- **Memory Usage:** The hybrid system requires more memory than traditional cryptographic systems due to the neural network layer. However, the model is designed to be lightweight, with minimal memory usage compared to deep learning models used in other domains.

- **Energy Efficiency:** Given that many low-resource systems operate on limited power (e.g., battery-operated IoT devices), we tested the energy consumption of the HNC model. The results show that while the neural network adds some overhead, the overall energy consumption remains relatively low compared to standard asymmetric encryption schemes like RSA.

Results and Discussion

The results indicate that the **Hybrid Neural Cryptography** model offers substantial improvements in terms of both security and performance:

- **Security:** The model effectively enhances the security of traditional cryptographic methods by applying neural network transformations, making attacks more difficult and increasing the overall complexity of the system.

- **Efficiency:** The HNC model provides a significant reduction in computational overhead compared to pure neural cryptographic systems and is highly suitable for low-resource systems. It strikes a balance between security and computational feasibility, making it appropriate for IoT devices, mobile applications, and other embedded systems.

Conclusion and Future Work

In this paper, we introduced the **Hybrid Neural Cryptography (HNC)** model, a novel cryptographic approach that integrates classical encryption techniques with the power of neural networks. Our experiments demonstrate that this hybrid model significantly improves both the security and efficiency of data encryption systems, particularly in resource-constrained environments. Future work will focus on optimizing the neural network architecture for better scalability, further enhancing the robustness of the system against adversarial attacks, and exploring the potential applications of the HNC model in real-world IoT and mobile security systems.

References

- Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*.
- Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 120-126.
- Zhang, C., & Wang, X. (2020). Neural network-based cryptographic systems: A survey and analysis. *International Journal of Information Security*, 19(5), 453-468.
- Liu, J., & Yang, C. (2018). Hybrid encryption scheme based on deep learning for secure communication in IoT. *IEEE Transactions on Industrial Informatics*, 14(5), 2111-2118.