BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 31 No. 1. 2026

UDC 621.391; 004.8; 004.426
DOI: 10.62660/bcstu/1.2026.24

ISSN: 2306-4412
E-ISSN: 2708-6070

Methodology development for training neural differential distinguishers and key recovery attack on single ciphertext

Xue Jiang*
Postgraduate Student
National University of Life and Environmental Sciences of Ukraine
03041, 15 Heroiv Oborony Str., Kyiv, Ukraine
https://orcid.org/0009-0000-1676-2331

Valerii Lakhno
Doctor of Technical Sciences, Professor
National University of Life and Environmental Sciences of Ukraine
03041, 15 Heroiv Oborony Str., Kyiv, Ukraine
https://orcid.org/0000-0001-9695-4543

Abstract. Key recovery attacks pose a critical threat to cryptographic systems, as these attacks directly compromise the security of encryption mechanisms. The purpose of this study was to develop a novel single-ciphertext key recovery method that addresses the limitations of traditional cryptanalysis techniques in scenarios with limited data availability. The methodology combined an enhanced B-C3-HSwish neural distinguisher with Bayesian optimisation guided by the Upper Confidence Bound (UCB) strategy. The neural distinguisher was improved through multiscale convolutional layers, activation function optimisation, and diversified input structures, while Bayesian optimisation mapped the key recovery problem to a high-dimensional search task using a Gaussian process surrogate model and the UCB acquisition function. Experimental results on the Speck32/64 encryption algorithm demonstrated a 56% success rate in 11 rounds of key recovery attacks, with data complexity ranging from 213.5 to 213.6. This result surpassed the 52.1% success rate obtained in the reference. The B-C3-HSwish neural discriminator has been improved, in particular by replacing the Relu activation function with Hard_swish, to increase its accuracy in single-pair input mode. This study not only offered a new, resource-efficient solution for key recovery in single-ciphertext scenarios, but also highlights the powerful potential of Bayesian optimisation as a promising paradigm for advancing cryptanalysis. The practical value of this work lies in its ability to provide a resource-efficient solution for key recovery in single-ciphertext scenarios, making it particularly suitable for real-world applications with constrained data. This approach also establishes Bayesian optimisation as a promising paradigm for advancing cryptanalysis and enhancing cryptographic security

Keywords: Bayesian optimisation algorithm; feature extraction optimisation; attack scenario; Speck32/64 algorithm; neural networks

INTRODUCTION

As the intersection of deep learning and cryptography has progressively deepened, the application of neural networks in cryptanalysis has garnered considerable attention. In the modern environment of rapid development of both cryptographic data protection methods and methods for breaking such protections, there is a growing need for new approaches to analysing the robustness of cryptographic algorithms. The application of deep learning in cryptanalysis is becoming particularly relevant, as it allows for the automatic detection of complex patterns and vulnerabilities that are difficult to detect using traditional methods. As a result, the application of neural networks in cryptanalysis is attracting increasing attention from researchers.

Article's History: Received: 23.06.2025; Revised: 21.01.2026; Accepted: 16.03.2026; Published: 08.04.2026

Suggested Citation:
Jiang, X., & Lakhno, V. (2026). Methodology development for training neural differential distinguishers and key recovery attack on single ciphertext. *Bulletin of Cherkasy State Technological University*, 31(1), 24-33. doi: 10.62660/bcstu/1.2026.24.

*Corresponding author

According to research by Z. Hou *et al.* (2021), an enhanced neural distinguisher was proposed to improve the performance of cryptanalysis. The authors' work significantly boosted the accuracy of the distinguisher in differential cryptanalysis by optimising network structures and input data formats. The authors highlighted that the integration of convolutional neural networks (CNNs) and attention mechanisms can effectively capture nonlinear features in ciphertext, thereby enhancing model performance. A. Gohr (2019) explored how deep learning techniques could be employed to attack the reduced-round Speck32/64 cipher. By incorporating deep neural networks, the author successfully enhanced the efficiency of the attack and demonstrated the potential of neural networks in cryptanalysis. The author concluded that deep learning not only accelerates cryptanalysis but also uncovers patterns that are challenging to detect using traditional methods. In particular, A. Gohr *et al.* (2022) investigated the effectiveness of using deep convolutional neural networks for differential cryptanalysis tasks. The authors focused on studying how neural networks can automatically learn to distinguish encrypted texts derived from certain known differential characteristics. The results of the study demonstrated that deep CNN models are capable of achieving high accuracy, even surpassing traditional statistical methods, particularly with fixed encryption parameters.

In the researchers' work, L. Zhang & Z. Wang (2022) conducted a comparative analysis of several modern architectures, including CNN, and ResNet, for use in differential cryptanalysis tasks. The main focus was on the effectiveness of model training when varying the amount of training data and the complexity of differential characteristics. The authors concluded that ResNet models have high stability and generalisation ability under limited training sets, demonstrating improved accuracy compared to other approaches. In contrast, S. Zhu *et al.* (2025) focused on adapting ResNet to analyse high-dimensional ciphertext, which is characteristic of modern block ciphers. The research covered network depth optimisation and the application of various normalisation mechanisms to avoid overfitting. As a result, the authors confirmed that ResNet not only maintains accuracy when working with large amounts of data, but also better copes with detecting weak differential characteristics in complex cryptographic constructions. Through experiments comparing various neural network architectures in differential cryptanalysis, the researchers concluded that residual networks (ResNets) perform exceptionally well when handling complex ciphertext, particularly in high-dimensional data environments.

The research by A. Benamira *et al.* (2021) delved into the application of machine learning in cryptanalysis. Experimental validation of the effectiveness of various machine learning models in cryptographic attacks revealed that, although neural networks perform well in cryptanalysis, the interpretability remains a pressing issue. The authors suggested that future research should focus on improving model interpretability to better understand the decision-making processes. A. Baksi *et al.* (2021) proposed the application of machine learning techniques to enhance differential distinguishers for lightweight ciphers. By integrating machine learning with traditional cryptographic analysis methods, the authors demonstrated significant improvements in the identification of differential characteristics, thereby advancing the efficiency of cryptanalysis for lightweight ciphers. The authors' work underscored the potential of machine learning in optimising cryptographic algorithms and highlighted its relevance in the evolving field of lightweight cipher security.

P. Auer (2020) explored the application of online learning in cryptanalysis. By introducing the Upper Confidence Bound (UCB) algorithm, the author proposed an online learning framework capable of dynamically adjusting model parameters to adapt to continuously changing ciphertext data. The research demonstrated that online learning can effectively enhance model performance in real-time cryptanalysis. D. Lin *et al.* (2022) proposed the application of the Bayesian optimisation algorithm in cryptanalysis. By combining Bayesian optimisation with neural networks, the authors successfully improved the efficiency of cryptanalysis and highlighted the importance of Bayesian optimisation in hyperparameter tuning. The research provided new insights into automated model optimisation in cryptanalysis.

Although these studies have made significant progress in the intersection of neural networks and cryptography, some aspects remain underexplored. For instance, existing research primarily focuses on differential cryptanalysis, while the application of linear cryptanalysis has received less attention. Additionally, how to generalise neural network optimisation strategies across different types of cryptographic algorithms remains an underexplored issue. These questions served as the focus of this study, aiming to provide new insights for the further application of neural networks in cryptanalysis. The aim of this study was to propose a new single-ciphertext key recovery method that overcomes the limitations of traditional cryptanalysis techniques in data-scarce scenarios.

MATERIALS AND METHODS

This research conducted an in-depth literature survey, focusing on the following aspects: network structure optimisation (different types of network structures, such as convolutional neural networks, residual networks, attention mechanism networks, etc., and the advantages in feature extraction and expression ability were studied); input data structure optimisation (discusses how to optimise the ciphertext to input mode, in order to better use the learning ability of the neural network to improve the performance of the differentiator); activation function and the normalisation technique (compare the different activation functions, and

normalisation methods affect the performance of model, to find the best combination to improve the accuracy of a differentiator).

In this study, A. Gohr (2019) attack approach was adopted. By assuming an s-round differential $\delta \rightarrow \Delta$ with a conversion probability of , denoted as Chosen Differential (CD), the γ-round distinguisher was extended to an $s+\gamma$ round distinguisher. Due to the probabilistic nature of the preset differential, it was required to generate approximately $c \times 2^{p}$ (denoted as n_{cts}) plaintext pairs with input differences of δ, where c is a small constant. Next, using the n_b neutral bits of the s-round CD, each plaintext pair was expanded into a plaintext structure containing n_b plaintext pairs. Since Speck32/64 does not perform key addition before the first nonlinear operation, the differential propagation characteristics of single-round encryption were not affected by the secret key. Therefore, the generated n_cts ciphertext structures could be decrypted at zero cost with a zero key and extended forward by one round to obtain the plaintext structure. Finally, all obtained plaintext structures underwent $1+s+\gamma+1$ rounds of encryption to obtain the ciphertext structure for the key recovery attack. After obtaining the high-accuracy differential distinguisher through the neural network, A. Gohr (2019) used the differential distinguisher of two rounds of 6 and 7 to carry out a key recovery attack against the 11 rounds of Speck encryption, which is the purpose of differential analysis. The symmetric cipher was broken as a whole by performing a key recovery attack through the differential distinguisher to obtain the real key.

The initial idea of extending rounds for decryption can be presented as a sequential operation with prepared differences and gradual addition of rounds to a known differential diffuser. First, a pair of plaintexts with a difference (0x211, 0xa04) was selected; after two rounds of encryption, this difference was converted into a high-probability difference (0x40.0) with a probability of 2^{-6}, which allowed it to be extended to the equivalent of 9 rounds based on a 7-round differential. Further, using Speck's property, according to which one round can be "added" without additional probability costs, the resulting 9-round differential was expanded to 10 rounds without loss. When restoring the key, the pairs of ciphertexts were decrypted one round back, obtaining N–1-round ciphertexts, and a differential distinguisher was applied to these results; thus, the 10-round analysis can be raised by one more round, achieving effective coverage of 11 rounds of encryption (Fig. 1).

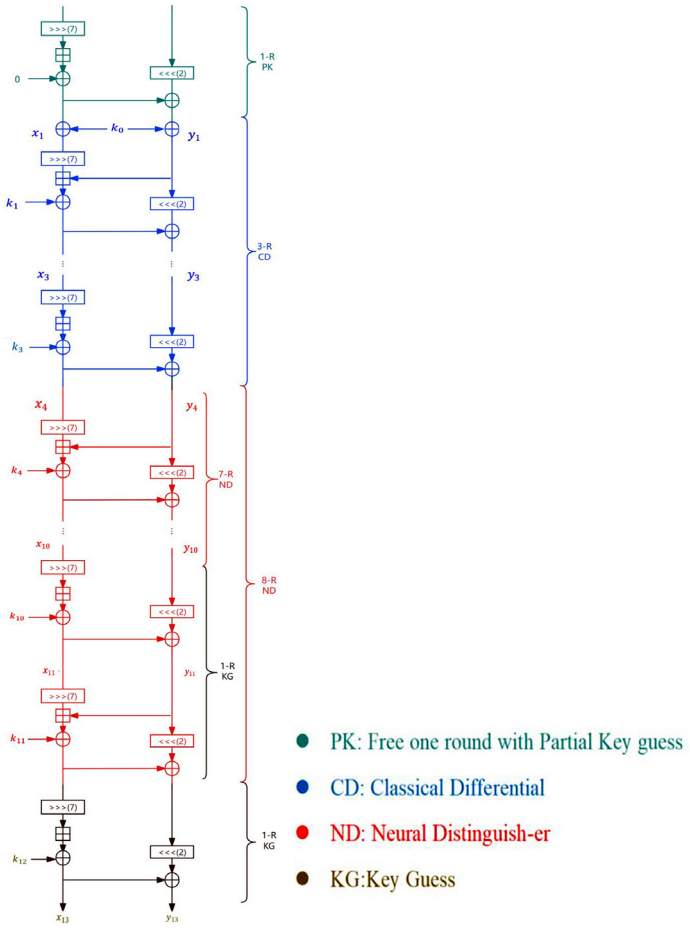

Figure 1. Round expansion process

Source: compiled by the authors

Since CD is probabilistic, only a 2^{-p} probability of plaintext structures satisfying the difference value δ will propagate to ciphertext structures with a difference of Δ after s rounds of differential propagation. This results in low computational efficiency. To accelerate the attack, A. Gohr (2019) introduced a UCB exploration-exploitation method for setting confidence interval upper limits. Each ciphertext structure was assigned a priority level, and before each key guess, the ciphertext structure with the highest priority was selected. This process was repeated n_{it} times to concentrate computational resources on the most likely correct ciphertext structures, thereby improving the attack efficiency.

The distinguisher's response to guessed subkeys depends on the Hamming distance from the true subkey. Therefore, the pre-generated incorrect key response file served as the basis for the Bayesian subkey search. By minimising the weighted Euclidean distance, the next candidate subkey was obtained. After selecting the most promising ciphertext structure C_i ($1 \leqslant j \leqslant n_{cts}$) candidate subkeys were randomly picked from the possible keys and entered into the Bayesian optimisation algorithm for n_{cand1} iterations. In each iteration, the candidate subkey $k_{1,j}$ ($1 \leqslant j \leqslant n_{cand1}$) was used to decrypt one round of the $(1+s+\gamma)$ round ciphertext structure C_i, obtaining a score $z_i^{k1,j}$ using the γ-round distinguisher, which was then combined into $v_{k_{1,j}}$. The response file selected the first n_{cand1} smallest Euclidean distance keys as the next guessed keys.

When searching for γ-round candidate subkeys using the Bayesian optimisation algorithm, the subkey $k_{1,j}$ and its score $v_{k_{1,j}}$ were stored. When $v_{k_{1,j}}$ exceeded the threshold c_1, the reciprocal second-round candidate subkey $v_{k_{2,j}}$ ($1 \leqslant j \leqslant n_{cand2}$) was used to decrypt one round to obtain the $1+s+\gamma-1$ round ciphertext structure, and the neural distinguisher and response file search were performed using the $\gamma-1$ round. If $k_{2,l}$ scored higher than the threshold c_2, the search was terminated, and k_1 and k_2 were returned for verification. Preset threshold – $C_{11}=10$, $C_{10}=10$. Before returning the guessed key, it is necessary to verify the search within a small range around it. Find subkey groups with a Hamming distance of 2 from $k_{1,j}$ and $k_{2,j}$, combine these groups to form ciphertext structures, and use the γ-round and $\gamma-1$ round distinguishers to score the resulting structures. Return the subkey corresponding to the best score, eliminating bit errors. If the verification improves, repeat the verification with the new best-guessed key.

In this paper, the standard-based UCB algorithm was used to assign priorities to each ciphertext structure according to its recommended sub-key score and access times and was combined with Bayesian optimisation to speed up the key recovery process. The key recovery attack was considered successful when the final returned subkey is correct and the second-to-last wheel key had at most 2-bit of error. The data were divided into training, validation, and test subsets (approximately 80/10/10) with independent random number generator seeds; classes were balanced, and bit representations were normalised according to the architecture (bit-plane or one-hot for CNN). The model used a binary loss function (binary cross-entropy), Adam optimiser ($\beta_1=0.9$, $\beta_2=0.999$, $\varepsilon=1e-8$) with a starting learning rate of $1e-3$, a mini-batch size of 256, and L_2 regularisation (weight decay $\approx 1e-5$); in full experiments, dropout 0.3 was applied in fully connected layers. Training lasted up to a maximum of 200 epochs with early stopping with a patience of 20 epochs on the validation metric (validation loss or AUC), and the training plan used ReduceLROnPlateau to reduce the learning rate in the absence of improvements. For stability, the results were obtained with three different random seeds, while checkpoints with the best validation values were saved and all key metrics (train/val loss, AUC, accuracy) were logged to ensure full reproducibility.

The complete key recovery attack protocol integrated a neural discriminator with Bayesian optimisation as a single procedure for searching and verifying candidates. Initially, a pool of 32 initial candidates was formed, which were obtained either by pre-ranking a large set of hypotheses by a neural network on a small subset of data, or by combining statistical heuristics and local mutations; for each candidate, partial decoding to an intermediate state (N–1 round) was performed, and these intermediate vectors were fed to the recogniser. The recogniser returned the probability *p_k* that the intermediate values correspond to the expected differential; for input into the optimisation procedure, *p_k* was aggregated (averaged or median across several batches) and, if necessary, converted to log-odds.

Bayesian optimisation worked with a surrogate Gaussian Process (RBF-kernel) model or TPE as a fallback and used the Expected Improvement acquisition function (alternatively UCB); optimisation was initiated with *n_init* ≈ 20 random evaluations and performed up to 500 iterations, with BO proposing the next candidate for evaluation at each iteration, obtaining *S_k* (target evaluation, e.g. – *logit*(*p_k*) or another aggregated quality function) and updating the surrogate. The stopping criteria included reaching the maximum number of iterations, no improvement within a specified patience period, or reaching the threshold *p_k* (e.g., ⩾ 0.99); final verification was performed for the top M candidates for *S_k* (e.g., $M=5$) by complete decryption. For statistical evaluation of the attack's reliability, $N=100$ independent runs were performed with different seeds and sets of text pairs, after which the success rate, the average number of decryption oracle calls, and the average time were recorded; standard statistical tests (t-test or bootstrap for confidence intervals) were used for comparisons between approaches.

RESULTS AND DISCUSSION

The network structure following the aforementioned improvements is designated as B-C3-HSwish (Jiang *et al.*, 2025), as illustrated in Figure 2. Specifically, the

module has been redesigned into a multi-branch parallel convolutional structure, which not only enhances the diversity of feature extraction but also improves the overall expressiveness of the network. Furthermore, the original ReLU activation function has been replaced with the Swish activation function, effectively mitigating the issue of "neuron death" that may arise due to ReLU's constant zero output in the negative region, while also enhancing the network's adaptability to different input data distributions.

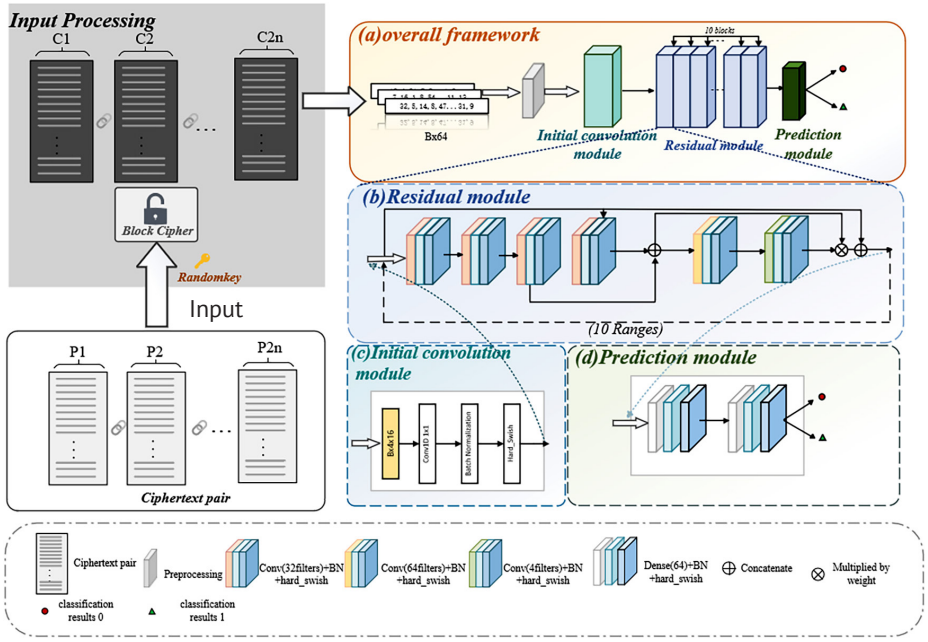

Figure 2. Overall structure diagram of the improved neural network B-C3-HSwish
Source: compiled by the authors

As a result of the literature analysis, it can be summarised, that the network structure after the aforementioned improvements is named B-C3-HSwish, as shown in Figure 2. Specifically, Module 2 has been redesigned into a multi-branch parallel convolutional structure, which not only enhances the diversity of feature extraction but also improves the overall expressiveness of the network. Moreover, the original Relu activation function has been replaced with the Swish activation function, effectively alleviating the issue of "neuron death" that may occur due to Relu's constant zero output in the negative region, while also enhancing the network's adaptability to different input data distributions. A model of a multiscale convolution enhancement scheme was proposed, which integrates the multiscale convolution method and network design principle of RegNet. 3 * 3. By utilising stacked convolutions and adding convolutional branches, the impact of these modifications on the distinguisher accuracy was investigated. This method significantly enhances the feature extraction function of the model.

Activation function improvement: activation function and normalisation can improve model expression ability, prediction accuracy, stability, and convergence speed in neural networks. Based on the improved distinguisher, this paper conducted experiments on the activation function and normalisation. In the residual structure, Relu makes the "residual" branch result non-negatively increasing, which affects the feature expression. To solve this problem, this paper adjusted the position of the activation function or replaced it with Swish, LeakyRelu, etc. After 5 rounds of discrimination tests of Speck encryption ciphertext pairs, Hard_swish was finally determined to replace Relu. Development of an encryption method based on multiple ciphertexts for input structure: By using random keys for one round of encryption, the diversity of input data was realised and the learning ability of the neural network for ciphertext features was enhanced. This significantly improved the generalisation ability and training efficiency of the model.

In this paper, a key search strategy based on Bayesian optimisation was adopted to improve the search accuracy and efficiency in the attack model. In the attack model, the distinguisher is regarded as a black box function, and the maximum value of the function is the candidate sub-key. In order to find the maximum value, the Bayesian optimisation method was used in the recovery process, which includes two core processes: the prior function and the collection function. The prior function was solved by Gaussian process regression, which was then solved by Bayesian inference. The acquisition function included an exploration-exploitation method, aiming to find the best sample set. That is, it first guessed a sub-key, decrypted the ciphertext using the sub-key, and obtained the result through the

black-box function. Then the candidate sub-keys were obtained by Bayesian inference using the values of the sampling points. The value of the sample point can be calculated by precomputing the sample set in the case that the wrong key is obtained. Compared with the traditional search strategy, the Bayesian optimisation method can quickly find the maximum value of the function and its corresponding parameters without knowing the function equation, so as to improve the accuracy and efficiency of key search.

The frequency of recovery attacks was set to 100, and 100 ciphertext combinations containing 11 rounds were constructed during each attack. The formation of these ciphertext structures was based on neutral bit-flipping technology. Specifically, each original ciphertext can be extended to 64 derivative ciphertexts by a 6-bit bit flipping operation, and the required 100 ciphertext structure groups were finally constructed by using this bit expansion mechanism. This design enabled a single base ciphertext pair to be systematically generated into a complete data structure containing 64 associated ciphertexts after bit transformation. The maximum number of iterations was set to 500 when searching the ciphertext structure. In each iteration, the UCB algorithm was used to select the ciphertext structure with the current highest priority, and the initial selection is the first ciphertext structure.

The decryption evaluation was carried out for the selected ciphertext architecture, and the Bayesian optimisation strategy was used for a total of five rounds of iterations. In each round of Bayesian optimisation, all ciphertext pairs were decrypted from 32 preset key candidates, and the key selection in the first iteration followed the random principle. After decrypting the ciphertext with these keys, the decryption results were scored by a specific distinguisher, and the score of each key and the average of the distinguisher scores were calculated according to (1):

$$S_k = \sum_{i=1}^{\frac{N}{k}} log_2\left(\frac{Z_i}{1-Z_i}\right), \quad (1)$$

where S_k – the score of the secret key; Z_i – the output signal given by the neural distinguisher for the *i*-th input sample.

For the baseline model, the group size *k* was set to 1. For the 32 initial keys, the Euclidean distance score between each initial key and each candidate key whose difference range was calculated, and the 32 keys with the smallest distance were screened out as the candidate keys for the next iteration – $[0,2^{16})$. By this method, each ciphertext structure only needs to try 160 candidate keys in the last round, and finally output the adopted keys and the corresponding scoring results. For candidate keys with scores exceeding the preset threshold $C_{11} = 10$, the Bayesian optimisation algorithm was further employed to filter out 160 candidate keys for 10 rounds and calculate the corresponding scores. If the score still exceeds the threshold $C_{10} = 10$, then the two selected keys were output as the final key guessing result. The success of key recovery was evaluated by calculating the bit difference between the guessed key and the real key. In this paper, five key recovery tests were carried out, each with 20 key recovery attacks, a total of 100 key recovery attacks, and the specific recovery situation is shown in Table 1.

Table 1. Results of key recovery attacks

Experimental Rounds	1	2	3	4	5
Number of successes in 20 recovery experiments	9	9	14	10	14
Data Complexity					
Success rate	45%	45%	70%	50%	70%

Source: compiled by the authors

As shown in Table 1, in 100 secret key recovery attacks using the B-C3-HSwish differential discriminator, the number of successful secret key recoveries reached 56 times, with a success rate of approximately 56%. The average data complexity of the attack was between $2^{13.5}$ and $2^{13.6}$. Compared to the success rate of 52.1% obtained by this method in reference, the B-C3-HSwish differential discriminator demonstrated higher accuracy in secret key recovery attacks. This was expected because the discriminator has higher discrimination accuracy in single-pair input mode.

Between 2020 and 2025, with the rapid development of deep learning technology, neural networks have been introduced into differential cryptanalysis, which has brought new research directions and possibilities to this field. Numerous studies have demonstrated that neural networks offer significant advantages in learning and recognising the internal differences characteristic of encryption algorithms. For instance, H. Niu *et al.* (2025) stated that A. Gohr was the first to propose the use of neural networks to construct a differential distinguisher, achieving superior results compared to traditional methods when analysing the lightweight block cipher Speck. Building on this foundation, the authors explored the potential of deep learning to further improve the effectiveness of differential distinguishers by introducing alternative network architectures and optimisation strategies. Despite these advancements, current neural network-based differential distinguishers still face several challenges. C. Jia *et al.* (2025) emphasised that classification accuracy remains a key concern, particularly when dealing with high-round

encryption, where the accuracy tends to decline significantly. Similarly, Y. Liu *et al*. (2024) observed that most existing distinguishers are only effective against a limited number of encryption rounds, and the performance becomes restricted when applied to more complex or higher-round cipher structures. In contrast to these challenges, this research demonstrated significant progress: by using an improved B-C3-HSwish discriminator and integrating Bayesian optimisation, a 56% success rate was achieved in key recovery attacks on 11 rounds of Speck32/64. This result not only confirmed the high accuracy of this model, but also directly addressed the problem of limited effectiveness at high rounds, setting a new benchmark for attacks in a single ciphertext scenario with low data complexity.

A study by N. Mukwevho & C. Chibaya (2020) focused on the design and evaluation of key scheduling algorithms (KSAs) in symmetric encryption systems. The research emphasised the critical role of KSAs in enhancing the security of block ciphers against cryptanalytic attacks. The proposed KSA architecture aimed to balance computational efficiency with cryptographic robustness, incorporating advanced algorithmic constructs to minimise processing time and resource consumption. Through statistical analyses, including entropy measures and non-linear complexity assessments, the study demonstrated the resilience of the proposed KSA against various cryptanalytic benchmarks, highlighting its superior performance and security enhancements over existing standards. This approach, on the contrary, evaluated the effectiveness of an attack rather than the robustness of the defence. Whereas studies such as N. Mukwevho & C. Chibaya (2020) focused on how to strengthen KSA to make key recovery more difficult, this work demonstrates that even with a potentially robust KSA (as in the case of Speck32/64), key recovery mechanisms based on deep learning and Bayesian optimisation can circumvent these barriers. In contrast, P. Yadav *et al*. (2025) conducted research, which highlighted the interdisciplinary nature of contemporary research and the importance of rigorous evaluation in complex systems. S. Dey *et al*. (2024) proposed HiSE, a hierarchical (threshold) symmetric-key encryption scheme aimed at enhancing scalability and decentralisation in symmetric encryption frameworks. This research addressed the distribution of secret keys to mitigate single points of attack, incorporating threshold mechanisms within a hierarchical structure. Evaluations showed that HiSE could support large-scale, decentralised applications while maintaining strong security assurances.

J. Chien (2020) addressed the fundamentals and advances in deep Bayesian mining and learning for natural language, exploring applications such as speech recognition, document summarisation, text classification, and machine translation, while emphasising the limitations of traditional deep learning in expressing semantic structures and optimising distribution functions. The tutorial highlighted advanced Bayesian and deep models, including hierarchical Dirichlet processes, recurrent neural networks, variational auto-encoders, and generative adversarial networks, and discussed the connections and effectiveness in handling symbolic and complex patterns in natural language. Variational inference and sampling methods were formulated to optimise complex models, and case studies were presented to illustrate applications in deep Bayesian mining, learning, and understanding. The tutorial concluded with future directions, aiming to introduce novices to deep Bayesian learning, motivate emerging topics in data mining and natural language understanding, and synthesise distinct lines of machine learning research. L. Lyu *et al*. (2022) improved upon existing framework by integrating deep learning into differential cryptanalysis, successfully identifying 8/9/10-round neural differential distinguishers for Simeck32/64 and recovering subkeys for 13/14/15-round Simeck32/64 with low data and time complexity, while also sharing parameter tuning insights for deep learning-assisted key recovery attacks. The application of the Bayesian optimisation algorithm in cryptanalysis is mainly reflected in automatic distinguisher design and key recovery attack, as evident in the work of B. Seok & C. Lee (2025). In the design of the automatic distinguisher, a Bayesian optimisation algorithm can be used to automatically search the optimal distinguisher parameters, so as to improve the performance of the distinguisher. In the key recovery attack, the Bayesian optimisation algorithm can be used to guide the key space search and infer the most probable key by evaluating the similarity between the encryption results of different key candidates and the actual ciphertext. Y. Yang *et al*. (2024) stated that although the Bayesian optimisation algorithm has shown great potential in cryptanalysis, its application in single ciphertext attacks is still relatively rare.

Findings made in this research highlighted the potential of neural networks and Bayesian optimisation in advancing cryptanalysis, particularly in scenarios with limited data. The proposed method not only improved the success rate of key recovery but also reduced computational complexity, making it a practical and reliable solution for single-ciphertext attacks. Furthermore, the study underscored the importance of integrating machine learning techniques with cryptographic analysis to address emerging challenges in the field.

CONCLUSIONS

This study addressed the key recovery problem in single-ciphertext attack scenarios by proposing a key recovery attack method based on the Bayesian optimisation algorithm and the UCB strategy. By optimising the structure and input representation of the neural network differential distinguisher and integrating the Bayesian optimisation algorithm to intelligently search the key space, the proposed method significantly improves the

efficiency and accuracy of key recovery. Experimental results demonstrated that the method achieved a 56% success rate in 11 rounds of key recovery attacks, with a data complexity with a wide range, which is notably superior to traditional methods. This achievement highlighted the method's effectiveness and reliability in single-ciphertext attack scenarios. The study also revealed the powerful feature extraction capability of neural network differential distinguishers in single-ciphertext scenarios, enabling the method to efficiently identify key patterns with limited data. Additionally, the Bayesian optimisation algorithm proved to be highly efficient in high-dimensional search tasks, significantly reducing the computational overhead of key recovery. These findings not only provided a new technical pathway for key recovery attacks but also opened up new directions for the application of the Bayesian optimisation algorithm in the field of cryptanalysis. The combination of neural networks and Bayesian optimisation represented a novel approach that bridges the gap between machine learning and cryptographic analysis, offering a robust framework for future research. Looking ahead, this method can be further explored and applied to other encryption algorithms, potentially extending its utility to a broader range of cryptographic systems. Integrating more advanced machine learning techniques, such as deep reinforcement learning or transformer-based models, could further enhance the method's performance and applicability. Additionally, future work could focus on optimising the algorithm's design to reduce data complexity and improve success rates in more challenging scenarios. Through continuous refinement and validation, this study provided a solid foundation for advancing key recovery techniques and contributed to the development of more secure cryptographic systems.

ACKNOWLEDGEMENTS

None.

FUNDING

This research was sponsored by the Bengbu University Key Natural Science Research Project (Grant No. 2024ZR03zd).

CONFLICT OF INTEREST

None.

REFERENCES

[1] Auer, P. (2020). Using upper confidence bounds for online learning. In *Proceedings of the 41st annual symposium on foundations of computer science* (pp. 270-279). Redondo Beach: IEEE. doi: 10.1109/SFCS.2000.892116.

[2] Baksi, A., Breier, J., Chen, Y., & Dong, X. (2021). Machine learning assisted differential distinguishers for lightweight ciphers. In *2021 design, automation & test in Europe conference & exhibition (DATE)* (pp. 176-181). Grenoble: IEEE. doi: 10.23919/DATE51398.2021.9474092.

[3] Benamira, A., Gerault, D., Peyrin, T., & Tan, Q. (2021). A deeper look at machine learning-based cryptanalysis. In A. Canteaut & F.-X. Standaert (Eds.), *Advances in cryptology – EUROCRYPT 2021* (pp. 805-835). Cham: Springer. doi: 10.1007/978-3-030-77870-5_28.

[4] Chien, J.-T. (2020). Deep Bayesian data mining. In *Proceedings of the thirteenth ACM international conference on web search and data mining (WSDM '20)* (pp. 865-868). New York: Association for Computing Machinery. doi: 10.1145/3336191.3371870.

[5] Dey, S., Leander, G., & Sharma, N.K. (2024). Improved key recovery attacks on reduced-round Salsa20. *Designs, Codes and Cryptography*, 93, 243-262. doi: 10.1007/s10623-024-01522-7.

[6] Gohr, A. (2019). Improving attacks on round-reduced Speck32/64 using deep learning. In A. Boldyreva & D. Micciancio (Eds.), *Advances in cryptology – CRYPTO 2019* (pp. 150-179). Cham: Springer. doi: 10.1007/978-3-030-26951-7_6.

[7] Gohr, A., Leander, G., & Neumann, P. (2022). An assessment of differential-neural distinguishers. *Cryptology ePrint Archive*, 2022, article number 1521.

[8] Hou, Z., Ren, J., & Chen, S. (2021). Improve neural distinguisher for cryptanalysis. *Cryptology ePrint Archive*, 2021, article number 1017.

[9] Jia, C., Chen, J., Li, X., Zheng, H., & Zhang, L. (2025). BallPri: Test cases prioritization for deep neuron networks via tolerant ball in variable space. *Automated Software Engineering*, 32(1), article number 29. doi: 10.1007/s10515-025-00498-5.

[10] Jiang, X., Li, M., Makulov, K., Lakhno, V., & Sahun, A. (2025). Enhanced neural differential distinguisher for Speck32/64 using attention mechanisms and multi ciphertext inputs. *Informatica*, 49(19), article number 7889. doi: 10.31449/inf.v49i19.7889.

[11] Lin, D., Chen, S., Li, M., & Hou, Z. (2022). The construction and application of (related-key) conditional differential neural distinguishers on KATAN. In A.R. Beresford, A. Patra & E. Bellini (Eds.), *Cryptology and network security* (pp. 203-224). Cham: Springer. doi: 10.1007/978-3-031-20974-1_10.

[12] Liu, Y., Yang, L., Chen, J., Wu, W., & Feng, Y. (2024). Matrix computation over homomorphic plaintext-ciphertext and its application. *Journal on Communication. Series: Tongxin Xuebao*, 45(2), 150-161. doi: 10.11959/j.issn.1000-436x.2024024.

[13] Lyu, L., Tu, Y., & Zhang, Y. (2022). Deep learning assisted key recovery attack for round-reduced Simeck32/64. In W. Susilo, X. Chen, F. Guo, Y. Zhang & R. Intan (Eds.), *Information security* (pp. 443-463). Cham: Springer. doi: 10.1007/978-3-031-22390-7_26.
[14] Mukwevho, N., & Chibaya, C. (2020). Dynamic vs static encryption tables in DES key schedules. In *2020 2nd international multidisciplinary information technology and engineering conference (IMITEC)* (pp. 1-5). Kimberley: IEEE. doi: 10.1109/IMITEC50163.2020.9334110.
[15] Niu, H., McCallum, G.B., Chang, A.B., Khan, K., & Azam, S. (2025). Exploring unsupervised feature extraction algorithms: Tackling high dimensionality in small datasets. *Scientific Reports*, 15, article number 21973. doi: 10.1038/s41598-025-07725-9.
[16] Seok, B., & Lee, C. (2025). A novel approach to construct a good dataset for differential-neural cryptanalysis. *IEEE Transactions on Dependable and Secure Computing*, 22(1), 246-262. doi: 10.1109/TDSC.2024.3387662.
[17] Yadav, P., Mittal, K., Gill, P.S., Mittal, A., Raghuraman, K., & Kaushik, J.S. (2025). Role of intravenous azithromycin as adjunctive therapy in children with acute encephalitis syndrome (AES): An open-label randomized controlled trial. *Cureus*, 17(7), article number e87387. doi: 10.7759/cureus.87387.
[18] Yang, Y., Xiong, X., Liu, Z., Jin, S., & Wang, J. (2024). High-performance encryption algorithms for dynamic images transmission. *Electronics*, 13(1), article number 131. doi: 10.3390/electronics13010131.
[19] Zhang, L., & Wang, Z. (2022). Improving differential-neural distinguisher model for DES, Chaskey, and PRESENT. *ArXiv*. doi: 10.48550/arXiv.2204.06341.
[20] Zhu, S., Li, L., Hu, Z., & Hu, Y. (2025). BCS: A neural distinguisher method based on differential propagation uncertainty of nonlinear components and network adaptability. *Physica Scripta*, 100(3), article number 035008. doi: 10.1088/1402-4896/adae63.

Розробка методології для навчання нейронних диференціальних розпізнавачів та атаки на відновлення ключа для окремого шифрованого тексту

Сюе Цзян

Аспірант
Національний університет біоресурсів і природокористування України
03041, вул. Героїв Оборони, 15, м. Київ, Україна
https://orcid.org/0009-0000-1676-2331

Валерій Лахно

Доктор технічних наук, професор
Національний університет біоресурсів і природокористування України
03041, вул. Героїв Оборони, 15, м. Київ, Україна
https://orcid.org/0000-0001-9695-4543

Анотація. Атаки на відновлення ключів становлять серйозну загрозу для криптографічних систем, оскільки вони безпосередньо ставлять під загрозу безпеку механізмів шифрування. Метою цього дослідження було розроблення нового методу відновлення ключів з одним шифрованим текстом, який усуває обмеження традиційних методів криптоаналізу в сценаріях з обмеженою доступністю даних. Методологія поєднувала вдосконалений нейронний дискримінатор B-C3-HSwish з байєсівською оптимізацією, керованою стратегією верхньої межі довіри (UCB). Нейронний дискримінатор було вдосконалено за допомогою багатомасштабних конволюційних шарів, оптимізації функції активації та диверсифікованих структур вхідних даних, тоді як байєсівська оптимізація відобразила проблему відновлення ключів на високорозмірне завдання пошуку за допомогою сурогатної моделі гауссового процесу та функції придбання UCB. Експериментальні результати щодо алгоритму шифрування Speck32/64 продемонстрували 56 % успішність у 11 раундах атак на відновлення ключа, причому складність даних коливалася від 213,5 до 213,6. Цей результат перевершує 52,1 % успішність, отриману в еталонному дослідженні. Нейронний дискримінатор B-C3-HSwish був вдосконалений, зокрема, шляхом заміни функції активації Relu на Hard_swish для підвищення його точності в режимі введення однієї пари. Це дослідження не тільки пропонує нове, ресурсоефективне рішення для відновлення ключів у сценаріях з одним шифрованим текстом, але й підкреслює потужний потенціал байєсівської оптимізації як перспективної парадигми для розвитку криптоаналізу. Практична цінність цієї роботи полягає в її здатності надати ресурсоефективне рішення для відновлення ключів у сценаріях з одним шифрованим текстом, що робить її особливо придатною для реальних застосувань з обмеженими даними. Цей підхід також встановлює байєсівську оптимізацію як перспективну парадигму для просування криптоаналізу та підвищення криптографічної безпеки

Ключові слова: алгоритм байєсівської оптимізації; оптимізація вилучення ознак; сценарій атаки; алгоритм Speck32/64; нейронні мережі