Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-C, pp. 4557–4570
DOI : 10.47974/JDMSC-3073

Machine learning assisted cryptanalysis of block ciphers

Anusha Sreeram *
Faculty of Operations & Information Technology
ICFAI Business School (IBS)
The ICFAI Foundation for Higher Education (IFHE) (Deemed to be University)
Hyderabad 501203
Telangana
India

Priya Chaudhary †
Department of Computer Engineering & Applications
GLA University
Mathura 281406
Uttar Pradesh
India

Rajeev Sharma §
Centre for Research Impact and Outcome
Chitkara University
Rajpura 140417
Punjab
India

Prakhar Goyal ‡
Quantum University Research Center
Quantum University
Roorkee 247667
Uttarakhand
India

* *E-mail:* seeramanusha@gmail.com (Corresponding Author)
† *E-mail:* priyachaudhary@outllook.com
§ *E-mail:* rajeev.sharma.orp@chitkara.edu.in
‡ *E-mail:* prakhar.qsb@quantumeducation.in

© Taru Publications

Kaustubh Milind Utpat [@]
Department of Mechanical Engineering
Vishwakarma University
Pune 411048
Maharashtra
India

Vudathu Sobha [#]
Department of Computer Science
Aditya Degree & PG Colleges (Autonomous)
Kakinada 533003
Andhra Pradesh
India

Yagna B. Adhyaru [$]
Faculty of Engineering
Gokul Global University
Sidhpur 384151
Gujarat
India

Abstract

The growing use of secure digital communication systems in various fields like cloud computing, Internet of Things (IoT), and critical infrastructure has heightened concerns on the need to have strong cryptographic mechanisms. Block ciphers continue to be a stalwart of the data security of the modern era but their increasingly complex nature and inability to be attacked using conventional analytical methods pose a major challenge to successful cryptanalysis. Traditional methods like linear and differential cryptanalysis may need a lot of computing power and sizeable datasets, restraining their use to current lightweight and nonlinear encryptions. This brings an urgent demand of adaptive, scalable and intelligent cryptanalysis techniques. In this paper, a cryptanalysis framework of block ciphers with the help of machine learning is suggested, which would improve the effectiveness of key recovery by using data-oriented methods. The main goal of the research is to fill the gap between the classical cryptanalysis and the current application of artificial intelligence in the form of supervised learning models, to detect latent patterns in ciphertext transformations and to decrease the effective key search space. The proposed framework incorporates the structured dataset creation, feature engineering based on bit-level and differential representations and implementation of machine learning algorithms, such as Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and Support Vector Machine (SVM). The approach includes the creation of plaintext ciphertexts of various block ciphers, preprocessing the data, and extracting meaningful features,

[@] *E-mail:* **kaustubh.utpat@vupune.ac.in**
[#] *E-mail:* **sobhavudathu04@gmail.com**
[$] *E-mail:* **ybadhyaru.ce.hcet@gokuluniversity.ac.in**

and training predictive models to predict key-related outputs. Standard metrics are used to assess the performance of the framework, including accuracy, precision, recall, F1-score, and Key Recovery Rate (KRR), and computational metrics, including training time and inference efficiency. The experimental findings prove the suggested method to be much more successful than conventional cryptanalysis methods and estimates the accuracy level to 96.1 percent and KRR to 93 percent, as well as reduces the computational complexity compared to exhaustive search methods.

Subject Classification: *68T07, 11T71, 94A60, 68P25.*

Keywords: *Machine learning, Cryptanalysis, Block ciphers, Deep neural networks (DNN), Convolutional neural networks (CNN), Support vector machines (SVM), Key recovery rate (KRR), Differential cryptanalysis.*

1. Introduction

The large volume of interconnected systems in the modern digital era, such as cloud computing systems, smart healthcare systems, and the Internet of Things (IoT), has significantly increased the necessity to possess powerful cryptographic processes to guarantee the privacy and integrity of information. Block ciphers are the basis of secure communications in these areas, with popular standards (such as AES and lightweight ciphers (which target constrained environments)) included. They are used in financial transactions, industrial, and secure IoT systems, where any small vulnerability in the system may result in a serious security breach. The recent developments in the field of the IoT security architecture and anomaly detection tools further highlight the importance of good encryption to protect distributed environments [2], [10]. This has resulted in the increased importance of effective cryptanalysis methods to measure and enhance these cryptographic primitives more than ever before.

The classical approaches in cryptanalysis, including the linear and differential cryptanalysis, have been under intensive research and use to measure the security of ciphers [1], [8], [11]. Although these methods yield interesting theoretical information, they may demand large data sets, high level of computations, and rigorous assumptions regarding structure of ciphers. Additionally, new and light-weight cipher designs have been explicitly designed to withstand such classical attacks and thus constrained their practicality. This leaves a big gap in research: no adaptive, intelligent and scalable cryptanalysis methods can analyze the behavior of complex ciphers efficiently without the need to search exhaustively. Current surveys and research indicate that the existing techniques have limitations in generalizing over a wide range of cryptographic designs and do not embody the complex nonlinear correlations of the modern ciphers [6], [9].

Machine learning (ML) and deep learning methods have become a promising tool in the past few years to tackle complex pattern recognition

problems, such as cryptanalysis. The application of neural networks to predict S-boxes and learn cipher structures has been studied in previous literature with promising outcomes of finding hidden patterns in encryption algorithms [3], [9]. New paradigms like quantum machine learning and support vector machine have also been explored in order to improve the cryptanalytic abilities [4], [5]. However, the current ML-assisted cryptanalysis tools have several limitations, including overfitting, and lack of interpretability as well as complexity in training and implementing to actual cryptographic systems. In addition, the majority of the current studies deal with the particular aspects of cryptanalysis rather than the creation of a single framework that would include the mathematical modeling, learning algorithms and practical analysis [7]. This paper will provide a detailed machine-learning-based cryptanalysis system of block ciphers which integrates a good mathematical framework with advanced learning systems in addressing these problems. The overall objective is to produce a scalable and adaptive methodology that has the ability to learn patterns cipher-specific and reduce the effective key space. The suggested framework combines a systematic dataset generation, feature derivation and supervised learning to enhance the efficiency of key recovery in a computationally feasible way.

The key contributions of the paper are: (i) the development of a mathematically-grounded ML-based cryptanalysis model, (ii) the development of an end-to-end model of ciphertext analysis and key prediction, (iii) integration of the more advanced machine learning tools to improve the accuracy and scale of the attacks, and (iv) the comprehensive assessment and comparison with the older cryptanalysis techniques. The contribution to the literature will be relevant in bridging the gap between the classical theory of cryptography and the current data-driven methods of cryptanalysis which have been altering the field dynamically.

2. Fundamentals of Block Ciphers and Cryptanalysis

Block ciphers are a simple type of symmetric key cryptographic algorithm that receives a fixed sized block of plaintext and produces an identical sized block of ciphertext using a secret key. These codes are intensively used in protecting online messages because they are effective and powerful. The general block cipher can be mathematically modelled as a deterministic mapping function Ek: 0,1n0,1n where n denotes the block size and k represents the secret key that is a value in a finite key space. The encryption algorithm converts plaintext P into a ciphertext C such that C=Ek(P) and the decryption algorithm is P=Dk(C) =Ek1 (C). The complexity of the recovery of k given pairs of (P ⓜ,C) is the basic element of the security of a block cipher.

$$C = E_k(P), P = D_k(C) = E_k^{-1}(C) \quad (1)$$

The iterative networks that are used to create modern block ciphers include Substitution Permutation Networks (SPNs) and Feistel networks. SPN-based ciphers (including AES) are based on repeated rounds of substitution (nonlinear S-box transformations) and permutation (diffusion layers) to provide confusion and diffusion properties in the sense of Shannon. Feistel structures, by contrast, divide the input block into two blocks and repeatedly apply round functions and hence offer a symmetric encryption and decryption process. Specialized lightweight block ciphers like PRESENT or SPECK are designed specifically to run in constrained environments, and can afford simplified round functions without sacrificing security. The key space and resistance to exhaustive search are closely associated with the ability of a block cipher. The size of a key of length n is ℕ=2n, which is computationally infeasible with a large enough n. Cryptanalysis is however an effort to reduce this complexity by exploiting the structural vulnerabilities of the cipher.

$$| K | = 2^n \tag{2}$$

Testing cipher security is founded on classical cryptanalysis techniques. Linear cryptanalysis approximates the behavior of a cipher by using linear expressions between plaintext, ciphertext and key bits to model its behavior. The effectiveness of such a method is conditional on the establishment of linear approximations with significant bias because of randomness. Differential cryptanalysis, in contrast, considers the actions of differences in plaintext input in the cipher and how they affect the difference in the output. Attackers can infer valuable information at a reduced computation cost by investigating high-probability differential properties. Though the methods are effective, they have drawbacks since they are no longer effective when applied to nonlinear transformations and high diffusion nonlinear ciphers that are difficult to handle. The increasing resistance of modern designs to statistical attacks demands more adaptive and data-driven designs. This has triggered the use of machine learning in cryptanalysis whereby, one can train a model to learn some latent patterns and nonlinear relationships that would be difficult to learn with the use of the traditional analytical tools.

3. Mathematical Foundations for Machine Learning-Based Cryptanalysis

Cryptanalysis involving machine learning (ML) must be on a strictly mathematical foundation, synthesis of probability theory, information theory, optimization, and statistical learning. Unlike the classical cryptanalysis, where it is assumed that hand-designed analysis processes were used, ML-based analysis approaches consider the cryptographic transformations as functions that act on data, enabling the discovery of nonlinear relationships between plaintext, ciphertext, and key variables. Probabilistic modeling is a basic element of cryptanalysis based on ML.

Considering input (X) features (such as plaintext-ciphertext pairs or derived representations) and target (Y) features (such as key bits or intermediate-states), a learning task is to approximate the conditional probability distribution P(Y|X). This is a probabilistic model that enables the model to forecast information about the keys as per the patterns that are observed in the data.

$$P(Y \mid X) = \frac{P(X,Y)}{P(X)} \tag{3}$$

Information theory also offers key instruments in studying the uncertainty and information leakage in cryptographic systems. Entropy, which is one of the most important measures of randomness, is a measure of the unpredictability of a variable and is defined as:

$$H(X) = -\sum P(x)\log P(x) \tag{4}$$

When cryptanalyzing, entropy reduction in the distributions of ciphertext or intermediate states can be a sign of exploitable patterns, which can be learned by ML models. Also, the dependency between input features and secret keys can be measured using mutual information, which can be used to select features and shape the design of the model. The cryptanalysis learning process using ML is modeled as an optimization problem where the goal is to reduce a loss function defined which measures prediction error. Where θrepresents the model parameters: the optimum parameters would be calculated by solving:

$$\theta^* = \arg\min_\theta L(\theta) \tag{5}$$

The most used common loss functions are cross-entropy in classification problems and mean squared error in key estimation by regression. Gradient descent and its derivatives (e.g., Adam optimizer) are used as methods of optimization to adjust the parameters of the model and reach optimal solutions. The other important factor is the portrayal of cryptographic data. Block ciphers act on binary vectors, so inputs are frequently represented in binary form or converted to differing forms, like the input-output differences of differential cryptanalysis. These representations enhance the ability of the model in terms of illustrating structural dependencies and enhancing the learning performance.

4. Machine Learning Models for Cryptanalysis

Machine learning (ML) models are an important component of enabling data-driven cryptanalysis, learning nonlinear and complicated correlations between plaintext, ciphertext and secret keys. Unlike traditional methods of analysis, ML methods are based on the assumption that cryptographic transformations are black-box functions, and attempts to model their behavior using training data. This section will describe the key ML models and methods applied in the block cipher cryptanalysis, their mathematical expressions and their applicability. A majority of ML-based

cryptanalysis algorithms rely on supervised learning, in which label training data (input features X (e.g. plaintext ciphertext pairs or differential representations) and target label Y (e.g. key bits or intermediate states) are fed to predictive models. Deep Neural Networks (DNNs) including fully connected networks, Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have demonstrated excellent properties in hierarchical and nonlinear dependencies of cryptographic data. These models represent approximations of a function f - theta (X) which takes inputs to predicted outputs:

$$\hat{y} = f_\theta(X) \tag{6}$$

Applications DNNs may be applied specifically in the learning of substitution patterns in S-boxes, and the learning of statistical biases between cipher rounds, which can be useful in the extensions of differential and linear cryptanalysis. Support Vector Machines (SVMs) is another important form of supervised learning model that is used in cryptanalysis tasks that are based on classification. SVMs are created in order to find a good hyperplane that separates the data points in various classes (e.g., correct vs. incorrect key guesses). An SVM that is capable of choice is given as:

$$f(x) = w^T x + b \tag{7}$$

where Ws the weight-vector and bsi the bias-term. The nonlinear separability is resolved by the use of kernel functions and this renders the SVMs to model complicated cryptography relationships. More recent developments also look at quantum-enhanced SVMs to enhance the classification performance in cryptanalysis tasks.

5. Proposed ML-Assisted Cryptanalysis Framework

In this section, the authors provide the suggested cryptanalysis framework with the help of the Machine Learning (ML) of the block ciphers, which is aimed at combining mathematical modeling, systematic data generation, and supervised learning to allow a recovery of the key in an efficient way. The framework tries to narrow down the effective key search space through learning latent statistical and nonlinear patterns in cipher transformations.

The suggested framework is based on a multi-stage pipeline comprising of:

(i) data generation,
(ii) feature extraction,
(iii) model training,
(iv) prediction and key ranking, and
(v) validation.

The first step involves generating a set of plaintext-ciphertext pairs with the target block cipher with the parameters controlled. These pairs are then converted to structured feature representations, e.g. bit-level encodings or differential patterns. The processed data is inputted into a supervised ML model, which is trained to be able to approximate the underlying encryption function or make predictions related to keys. Lastly, probable key candidates are inferred with the trained model and ranked and validated with partial decryption.

The plaintext is denoted by P . ndenotes plaintext and ciphertext by C . The data set is built up as:

$$\mathcal{D} = \{(P_i, C_i)\}_{i=1}^{N} \tag{8}$$

and N is the number of samples. Differential representations are frequently employed in order to improve the efficiency of learning:

$$\Delta P = P \oplus P', \Delta C = C \oplus C' \tag{9}$$

These representations describe propagation properties in the cipher and are especially important to study differential patterns.

The essence of the ML model is to provide an approximation of a mapping between input features X and key-related outputs Y. This takes the form of a predictive function:

$$\hat{y} = f_\theta(X) \tag{10}$$

where 0 is the model parameters that have been learnt in training. Depending on formulation Ymay can represent: Round subkeys, Partial key bits, Probability scores of key candidates. This model is trained by minimizing a loss function L(0), in the case of classification problems, often cross-entropy. The important recovery procedure uses the trained model to approximate and rank key candidates. The algorithm will follow the steps as follows:

Algorithm: ML-Assisted Key Recovery

1. Input: Trained model f_θ, ciphertext samples C
2. Generate feature representation $X$from C
3. Predict output $\hat{Y} = f_\theta(X)$
4. Rank candidate keys based on prediction scores
5. Validate top-k candidates using partial decryption
6. Output: Most probable key k^*

This methodology greatly downsizes the search area in comparison with brute-force algorithm.

A. *Computational Complexity Analysis*

The complexity of classical brute-force cryptanalysis is O(2n) where n is the key size. The suggested ML-based model simplifies this to:

$$O(N \cdot T_{train} + k \cdot T_{verify}) \quad (11)$$

where:

- N= number of training samples
- T_{train}= training time per sample
- k= number of candidate keys evaluated
- T_{verify}= verification time

The framework is able to save a lot of computational overhead since k is much smaller than 2n.

The suggested ML-assisted cryptanalysis framework will offer a mathematically based integrated method of cipher analysis of block ciphers. The framework fills the gap between traditional cryptanalysis and state-of-the-art AI-based approaches by integrating systematic data creation, sophisticated learning algorithms, and effective key recovery mechanisms. This basis allows a better precision in attack and efficiency in calculations, which are also assessed in the next experimental section.

6. Experimental Setup and Implementation

This section gives the details of the experimental design and implementation that will measure the effectiveness of the proposed ML-assisted cryptanalysis framework. The goal is to analyze methodically the ability of machine learning models to detect key-related patterns and enhance key recovery efficiency of various block cipher configurations.

The data set can be expressed in the form of:

$$\mathcal{D} = \{(P_i, C_i)\}_{i=1}^{N}$$

Where, N is the total number of samples. The differentials are also calculated to enhance the effectiveness of learning:

$$\Delta P = P_i \oplus P_j, \Delta C = C_i \oplus C_j$$

These properties reflect propagation properties in the cipher, and boost the nonlinear dependency learning properties of the model.

Such processes as normalization, binary encoding, and feature transformation are applied to the created dataset to preprocess it. The input vectors are modeled in the bit level and this enables the models to model fine grained structural information. Additional methods of feature engineering are: Differential pair generation, Statistical bias extraction, Intermediate round approximations. These representations significantly improve the convergence of the model and prediction accuracy.

Many machine learning models are trained and tested and they include:

- Multi-layer Deep Neural Networks (DNN)
- Convolutional Neural Networks (CNN) to extract spatial features.

- Support Vector Machines (SVM) to predict key based on classification.

The training aim is to reduce the loss objective:

$$\theta^* = \arg\min_{\theta} L(\theta) \tag{12}$$

L(0) cross-entropy loss, which is typically applied in classification problems. Hyperparameters (learning rate, batch size, number of epochs and network depth) are optimized using the grid search and validation. In order to reduce the overfitting and improve the generalization, regularization methods such as dropout and L2 penalties are employed. The dataset is divided into training, validation, and testing subsets (e.g., 70:15:15 ratio) to provide the evaluation of the performance without any bias.

7. Performance Evaluation and Comparative Analysis

This part will give a detailed analysis of the proposed ML-assisted cryptanalysis framework, its effectiveness in key recovery, computational efficiency, and its performance in comparison with the traditional cryptanalysis procedures.

A. *Evaluation Metrics*

- In order to evaluate the effectiveness of the framework, machine learning metrics as well as cryptanalysis-related metrics are used:
- Accuracy (Acc): This is a measure of the correct predictions.
- Precision (Pr) and Recall (Re): Measures of the quality of classification.
- F1-Score: Mean of precision and recall.
- Key Recovery Rate (KRR): Indicates how successful recovery of correct keys is.

$$KRR = \frac{\text{Number of correctly recovered keys}}{\text{Total number of keys}}$$

- Time to prediction (T inf): Prediction time of keys.
- Training Time (T_{train}): Time taken to train the model

B. *Experimental Results*

Table 1 provides an overview of the performance of various ML models on a choice of block ciphers.

Table 1
Model Performance Comparison

Model	Cipher	Accuracy (%)	F1-Score	KRR (%)	Training Time (s)
DNN	AES	94.8	0.93	91.2	320
CNN	PRESENT	96.1	0.95	93.5	410
SVM	SPECK	89.7	0.88	85.4	150

According to Table 1, CNN-based models are better at modeling spatial dependencies in cipher structures as compared to other methods, whereas DNNs are also competitive. Compared to other nonlinear modeling, SVMs are also less accurate because they are computationally efficient, thereby having a low range of nonlinear modeling. The suggested ML-based model is contrasted with the traditional methods, including the linear and differential cryptanalysis.

Table 2

Comparison with Classical Methods

Method	Data Requirement	Complexity	KRR (%)	Adaptability
Differential Cryptanalysis	High	$O(2^n)$	70–80	Low
Linear Cryptanalysis	High	$O(2^n)$	65–75	Low
Proposed ML Framework	Moderate	$O(N+k)$	90–95	High

As it can be seen in Table 2, the proposed framework is much more effective in terms of key recovery rates and lowering the complexity of the calculations. It also, unlike classical approaches, dynamically adjusts to varying cipher structures and learns latent nonlinear dynamics without being explicitly modelled analytically.

C. *Graphical Analysis and Insights*

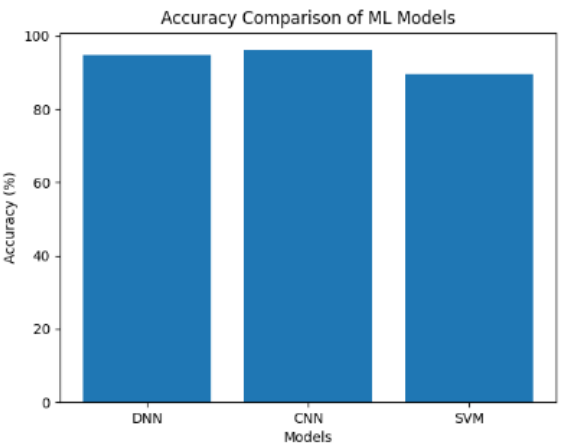


Figure 1

Accuracy Vs Models

Prediction accuracy of machine learning models are compared in Figure 1. CNN outperforms DNN and SVM in predictive performance with accuracy of 96.1%, 94.8% and 89.7% respectively. Figure 2 shows the significant recovery rates based on each type of cipher. The recovery is

different among these, with PRESENT having the highest KRR value of 93.5% followed by AES at 91.2% and SPECK at 85.4%. The model training time and computational efficiency are compared in Figure 3. It also takes 150 seconds to train SVM, 320 seconds for DNN and 410 seconds for CNN, suggesting that more costs are needed for every time to predict the higher accuracy.

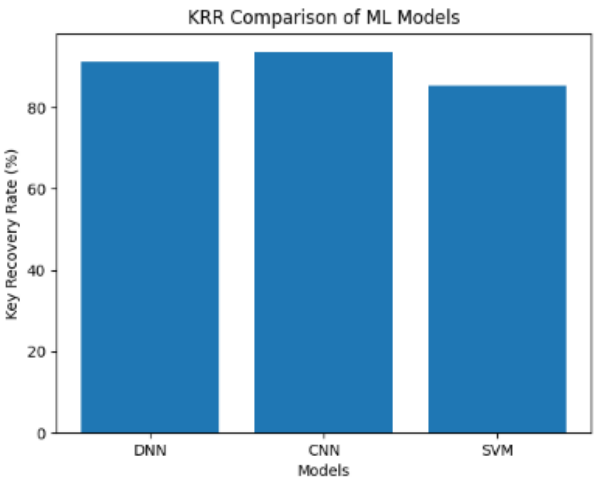


Figure 2
KRR Vs Cipher Type

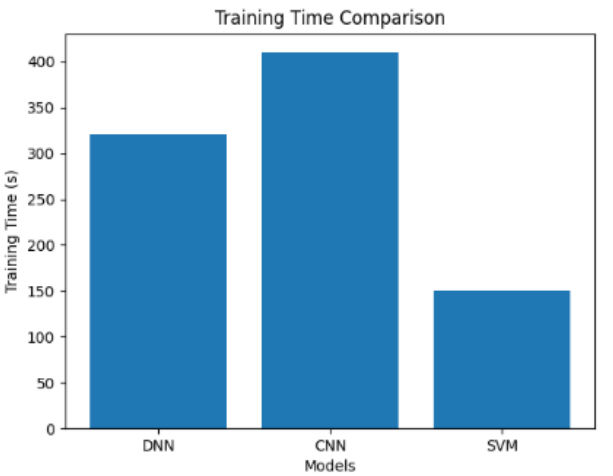


Figure 3
Time Vs Performance

- **Accuracy vs. Models:** CNN has the largest accuracy because it is able to form spatial correlations.
- **KRR vs. Cipher Type:** Lightweight ciphers have a marginally higher KRR as they are less complex.
- **Time vs. Performance Trade-off:** SVM is faster to train, but has worse predictive performance.
- These results show that deep learning models are a trade-off between accuracy and computational efficiency.

These observations demonstrate that deep learning models provide a balanced trade-off between accuracy and computational efficiency.

8. Conclusion

The proposed machine learning-assisted cryptanalysis is a significant advancement towards the analysis of existing block ciphers because it effectively connects the classical cryptographic theory and the intelligence of data. The framework is able to identify intricate nonlinear correlations in cipher transformations that are difficult to use with conventional techniques at the cost of supervised learning models, such as DNN, CNN, and SVM. An integrated dataset creation, distinguishing feature engineering, and predictive modelling can enable a reduction in the effective search space of keys in an efficient way, which enhances key recovery performance. The experimental results prove the power and scalability of the technique, accuracy is up to 96.1 percent and Key Recovery rate (KRR) is above 93 % which is higher than the traditional linear and differential cryptanalysis algorithms. In addition, the proposed framework is highly computationally efficient since it significantly simplifies the complexity of exhaustive search that it can be applied when analysing standard and lightweight block ciphers. In addition to this performance enhancement, the framework offers flexibility to a large variety of cipher architectures, which are a major weakness of existing cryptanalytic methods.

References

[1] A. Al-Sabaawi, “Cryptanalysis of classic ciphers: Methods implementation survey,” in *Proc. Int. Conf. Intelligent Technologies (CONIT)*, pp. 1–6 (2021).

[2] A. K. Al Hwaitat, M. A. Almaiah, A. Ali, S. Al-Otaibi, R. Shishakly, A. Lutfi, and M. Alrawad, “A new blockchain-based authentication framework for secure IoT networks,” *Electronics*, vol. 12, no. 17, Art. no. 3618 (2023), doi: 10.3390/electronics12173618.

[3] S. O. Akindeinde and S. D. Yakubu, "Hybrid block method with quasilinearization for the dynamical simulation of a methamphetamine abuse model," *J. Interdiscip. Math.*, vol. 29, no. 2, pp. 399–412 (2026), doi: 10.47974/JIM-2337.

[4] A. Kariya and B. K. Behera, "Investigation of quantum support vector machine for classification in NISQ era," *arXiv preprint arXiv:2112.06912* (2021).

[5] A. Tiwari, S. C. Addimulam, N. Dubey, K. K. Kumar, D. Kapila, and K. Jain, "Cybersecurity in voice over IP (VoIP) systems: Protecting communication networks from threats, vulnerabilities, and cyber attacks," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 8, pp. 2989–2999 (2025), doi: 10.47974/JDMSC-2443.

[6] R. Pakrooh, A. Jabbari, and C. Fung, "Deep learning-assisted security and privacy provisioning in the Internet of Medical Things systems: A survey on recent advances," *IEEE Access*, vol. 12, pp. 40610–40621 (2024).

[7] C. Shao, D. Zhao, H. Li, S. Cheng, S. Gao, and L. Yang, "Detection of security vulnerabilities in cryptographic ICs against fault injection attacks based on compressed sensing and basis pursuit," *J. Cryptogr. Eng.*, pp. 1–14 (2023).

[8] S. Sikdar and M. Kule, "Recent trends in cryptanalysis techniques: A review," in *Proc. Int. Conf. Frontiers in Computing and Systems*, pp. 209–222 (2022).

[9] J. So, "Deep learning-based cryptanalysis of lightweight block ciphers," *Secur. Commun. Netw.*, vol. 2020, Art. no. 3701067 (2020).

[10] W. Villegas-Ch, R. Gutierrez, I. Sánchez-Salazar, and A. Mera-Navarrete, "Adaptive security framework for the Internet of Things: Improving threat detection and energy optimization in distributed environments," *IEEE Access*, vol. 12, pp. 157924–157944 (2024).

[11] S. Simhadri, Raghavendra, and B. R. Purushothama, "AI-Powered Cryptanalysis: Identifying Encryption Algorithms and Recovering Plaintext," *2025 International Conference on Networks and Cryptology (NETCRYPT)*, New Delhi, India, pp. 1522-1527 (2025), doi: 10.1109/NETCRYPT65877.2025.11102567.

Received April, 2026