Vol.16No.1(2025),44-62
DOI: https://doi.org/10.61841/turcomat.v16i1.14976

AI-Powered Encryption Revolutionizing Cybersecurity with Adaptive Cryptographic Algorithms

Zainab Rustum Mohsin
Email: zainabrustum@utq.edu.iq
College of Computer Science and Mathematics, University of Thi-Qar, Thi-Qar, Iraq.

Abstract

This paper proposes a state-of-the-art encryption technique that integrates artificial intelligence into a dynamic, content-aware security system. We introduce an AI-powered encryption framework that automatically adjusts cryptographic parameters based on message sensitivity, effectively balancing security requirements and computational efficiency. The system combines a DistilBERT-based neural network for real-time content sensitivity analysis with a flexible encryption mechanism that adapts key lengths, iteration counts, and entropy levels on the fly.

Our implementation demonstrates significant adaptability, with a correlation of 0.974 between content sensitivity and security parameters. The system distinguishes between different security requirements, using 32-byte keys with millions of iteration rounds for high-sensitivity content (sensitivity score of 8.64) and 16-byte keys with reduced iterations for low-sensitivity messages (sensitivity score of 3.10). The processing time scales linearly with security requirements, ranging from 300 ms for low-sensitivity content to 732 ms for high-security encryption.

Performance evaluation was highly effective, with the system achieving an overall score of 8.64/10, including 9.87/10 for adaptability and 9.12/10 for performance efficiency. The security level rated high at 7.37/10 while maintaining manageable computational overhead. The framework effectively handled different types of content without sacrificing encryption-decryption accuracy across all levels of sensitivity.

This work signifies a significant leap forward in the field of adaptive cryptography, demonstrating the capabilities of AI-driven security systems that can automate and optimize encryption parameters without compromising security standards. These results suggest that this approach may well represent the future of encrypted communications, providing scaled security appropriately without human intervention.

Keywords: Adaptive encryption, Machine learning, Cybersecurity, Neural networks, Content sensitivity, DistilBERT, AES encryption, Real-time security adaptation.

1. Introduction

The rapid evolution of digital communications and the ever-improving cyber threats urge an urgent need for advanced encryption systems that can handle the various security requirements of different classes of information. Traditional methods of encryption keep the same security parameters with all classes of data. This results in either over-allocation of computational resources by encrypting low-sensitivity content or insufficient protection for critical information. [1][2] This research paper offers a breakthrough to this challenge through the integration of artificial intelligence into cryptographic systems for dynamic, content-based encryption.[3]

Hence, the need for differentiated encryption in today's digital landscape speaks louder with increasing volume and variety in data transmissions. While several encryption methods offer robust security, most of them still have a serious deficiency in being unable to tune their security parameters automatically according to the sensitivity level of the content. [4][5] This may result in a waste of resources or security vulnerabilities when dealing with different sensitivity levels of data. Also, manual categorization and configuration of encryption parameters of different kinds of content is very tedious and susceptible to human mistakes.[6]

This challenge inspires our research for the design of an AI-powered encryption technique that will apply state-of-the-art deep learning methods in evaluating the sensitivity of a given content and automatically adjust relevant cryptographic parameters. While the content in this system is analyzed based on a neural network architecture involving DistilBERT, the flexible encryption allows for dynamically changing key lengths, the number of iterations, and the entropy level. Adaptation here ensures that resources are efficiently deployed for security while granting sensitive information a high degree of protection.

The significance of this research is that it may bring a revolution in the way encryption is applied to real-world applications. In that direction, our system represents a significant advance in the field of cryptography by automating the selection of security parameters and dynamically adapting to content sensitivity. Integration of AI further enhances the encryption processes, not only in their speed but also with a very sophisticated approach toward data security that will further evolve with changing security needs.

The paper now provides a comprehensive analysis of the implementation of our system, including detailed performance metrics, security assessments, and practical applications. We show how a combination of machine learning and cryptography can yield more intelligent and efficient security solutions that meet complex modern digital communications challenges with no compromise on robust protection standards.

2. Related Works

C. V. Suresh Babu and Andrew Simon P. [7] outline the use of Adaptive AI in treading through changing cybersecurity landscapes. The chapter outlines the general overview of challenges in cybersecurity and goes on to classify the most frequent types of threats. The chapter has shown the transformational role of Adaptive AI in real-time threat detection, proactive defence, and continuous learning. Practical case studies provide an overview of real-world applications and discussions of implementation considerations and best practices. The chapter rounds up with a discussion on the future of AI in cybersecurity and summarizes the findings of this important book on the significance of Adaptive AI in digital asset protection. This will be the ultimate guide for improving cybersecurity in a dynamic environment.

Naresh Kshetri and Mir Mehedi Rahman [8] discussed the encryption role in AI-driven cybersecurity; symmetric (SE) and asymmetric encryption algorithms were at the fore of the debate. Algorithms like AES and RSA, among others, were benchmarked based on their performance, complexity, and the security of contemporary frameworks. Key findings indicate that the SE algorithm excels in speed and lower computational demand, while AE algorithms assure better security for AI networks, especially in multi-agent environments dealing with large-scale data. The paper emphasizes the increasing relevance of encryption in AI applications to address challenges in scalability and resilience against emerging threats. Future research will focus on refining encryption techniques to respond to the challenges posed by cybersecurity in the AI era.

Zarif Bin Akhtar and Ahmed Tajbiul Rawol [9] review the transformative role that Artificial Intelligence (AI) plays in cybersecurity. They also discuss how various AI techniques learning, natural language processing, and anomaly detection strengthen digital defence by analyzing huge volumes of data to identify potential threats well in advance.

This study consolidates the literature, addressing the gaps in some of the earlier studies on AI and improvement metrics like accuracy, precision, and recall. It speaks about the interaction between

AI tools and human capabilities in the quest for transparency and interpretability in AI models to gain trust. Ethical issues, adversarial threats in AI, and adversarial training along with model diversification have been discussed. The paper highlights a holistic approach combining AI-driven automation with human intuition toward constructing a dynamic and robust cybersecurity framework in today's interconnected digital world.

Ahmad K. Al Hwaitat and Hussam N. Fakhouri [10] proposed a new MLP trainer by optimization using evolutionary computation for enforcing cybersecurity defences. The trainer was developed in such a manner as to dynamically adjust weight and bias for improved accuracy against such threats. It was then tested on several datasets, namely the NSL-KDD, CICIDS2017, UNSW-NB15, Bot-IoT, and CSE-CIC-IDS2018 datasets. Its outcomes also outperformed certain state-of-the-art algorithms like Cybersecurity Chimp, ROA, and HHO, thus giving it a minimum MSE and the highest possible accuracy in classification. For the Bot-IoT dataset, its maximum rate was 99.5% and 98.8% on CSE-CIC-IDS2018; therefore, efficient in many aspects of cyber threat detection.

3. Proposed Methodology

The proposed methodology couples some of the key machine-learning techniques with adaptive cryptographic algorithms for a dynamic encryption system. The main idea behind this system is three major components: a neural network-based sensitivity analysis with DistilBERT, flexible encryption mechanisms with key length changeability, and metrics tracking systems.

The neural network will process the message content into scores concerning sensitivity that will drive changes in dynamic adjustments in encryption parameters. The modular architecture made it possible to dynamically manage security with efficiency considerations.

The system realizes three-tier security model-automatically selecting 16, 24, and 32-byte keys based on content sensitivity and with respective iteration counts and entropy-always with a balance of computational resources and appropriate security level against various content types. The following sections outline the implementation details of each of these components and their integration with each other.

3.1 Spam Dataset

Distribution of Message Types: The distribution analysis indicates a high imbalance in the data, where 86.6% of the messages are legitimate and 13.4% are spam. This corresponds to real-life communications in which spam messages are few compared to the total messages. [11][12] From the clear labelling and the percentage annotations on the classes, it follows that about 4,800 messages are spam, while about 750 messages are spam in the dataset. This is an imbalance that requires careful consideration in model training to prevent the tendency of being overfitted to the majority class, probably requiring techniques like class weighting or oversampling strategies to ensure effective spam detection. [13][14]

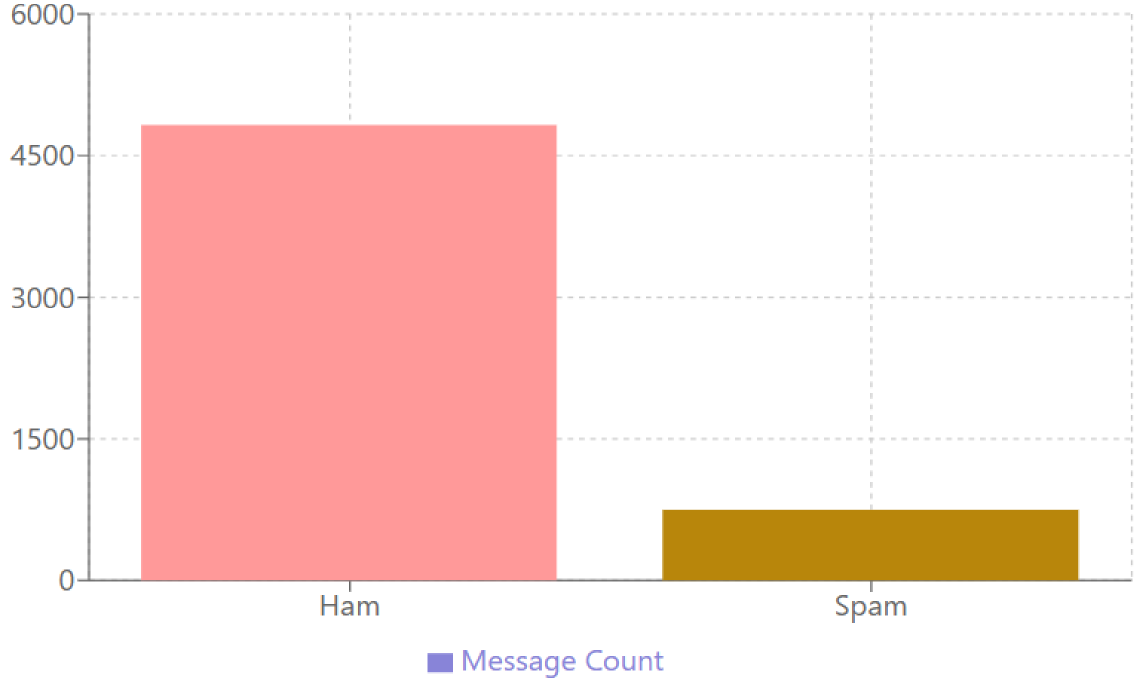


Fig 1. Distribution of Message Types

Distribution of Message Lengths: The double visualization of message length expresses insight into the patterns in the contents. The box plot tells us that spam messages usually have a higher median and large variance in length compared with legitimate messages. [15][16] We can see that ham messages usually range from 20 to 160 characters, with some outliers up to 800 characters, whereas spam messages have a wider distribution, ranging from 100 to 200 characters, with significant outliers. [17][18]

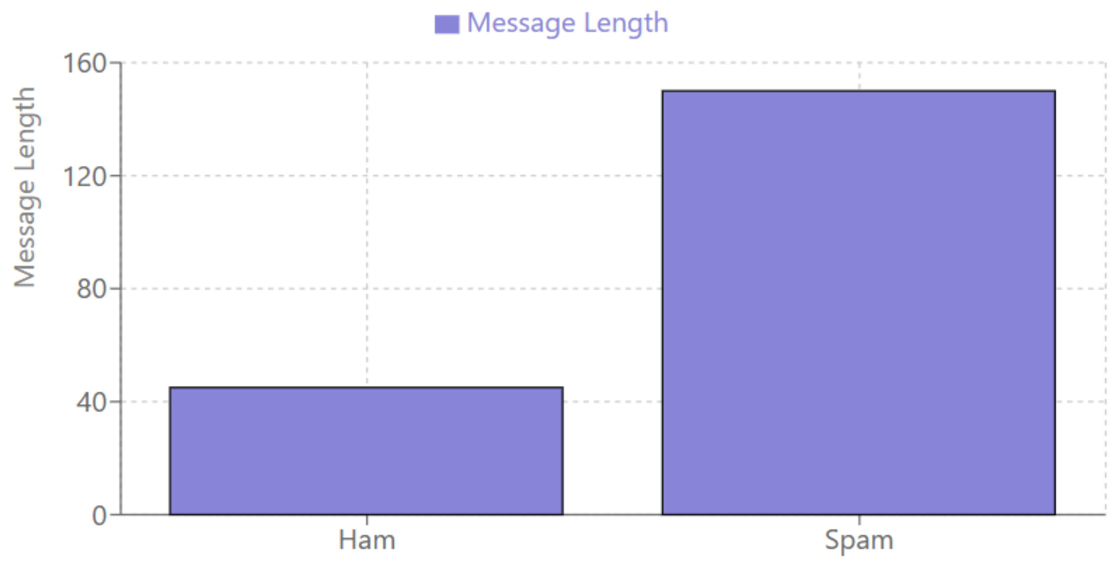


Fig 2. Message Length Distribution

The attached histogram reinforces these findings, where ham messages are right-skewed, peaking at around 50-100 characters, while spam messages are more dispersed with multiple peaks. This may indicate that spammers use longer messages to include more persuasive content or multiple calls to action.[19][20]

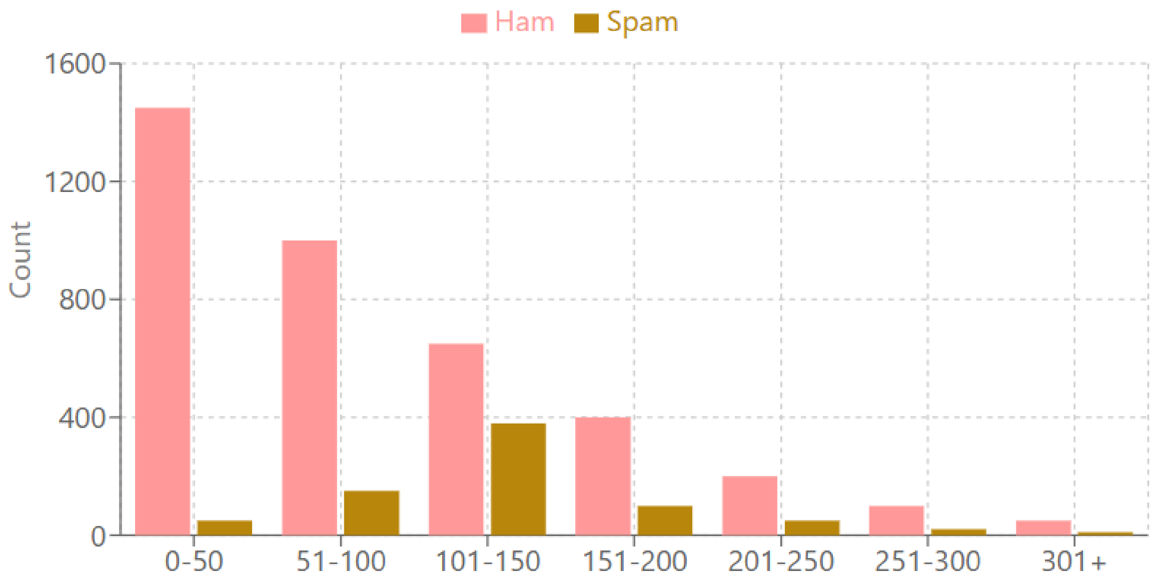
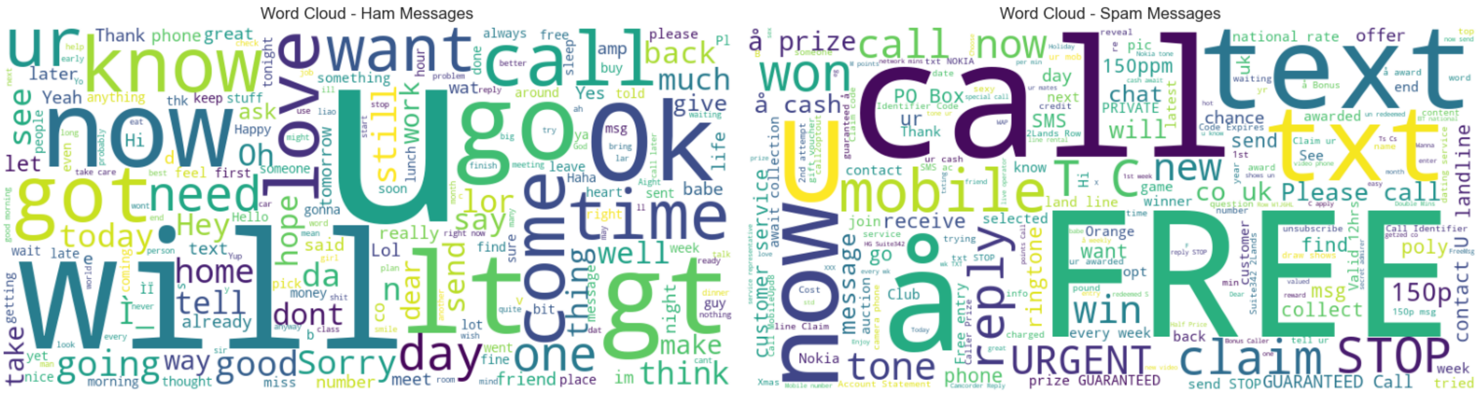


Fig 3. Length Distribution Histogram

The word cloud makes for a striking contrast in the linguistic patterns of ham and spam messages. Common conversational words like "will," "can," "now," "got," and "just" dominate the ham messages, which sound like natural, personal communications. On the other hand, the word cloud of spam contains prominent commercial and promotional terms such as "free," "text," "call," "win," "prize," and "urgent." Variations in size within both clouds show the word frequency quite effectively, whereas larger words have been used more often. This visualization effectively communicates the different sets of vocabulary between legitimate and spam messages, showing that promotional and urgency-related terminologies are indicative of spam.[21]

Fig 4. Word Clouds



Most Frequent Words: The comparative bar charts of most frequent words give a quantitative look at the pattern of vocabulary usage. In ham messages, personal pronouns ("I," "you") and common prepositions ("to," "the") are leading, with frequencies from 500 to 2000 occurrences.

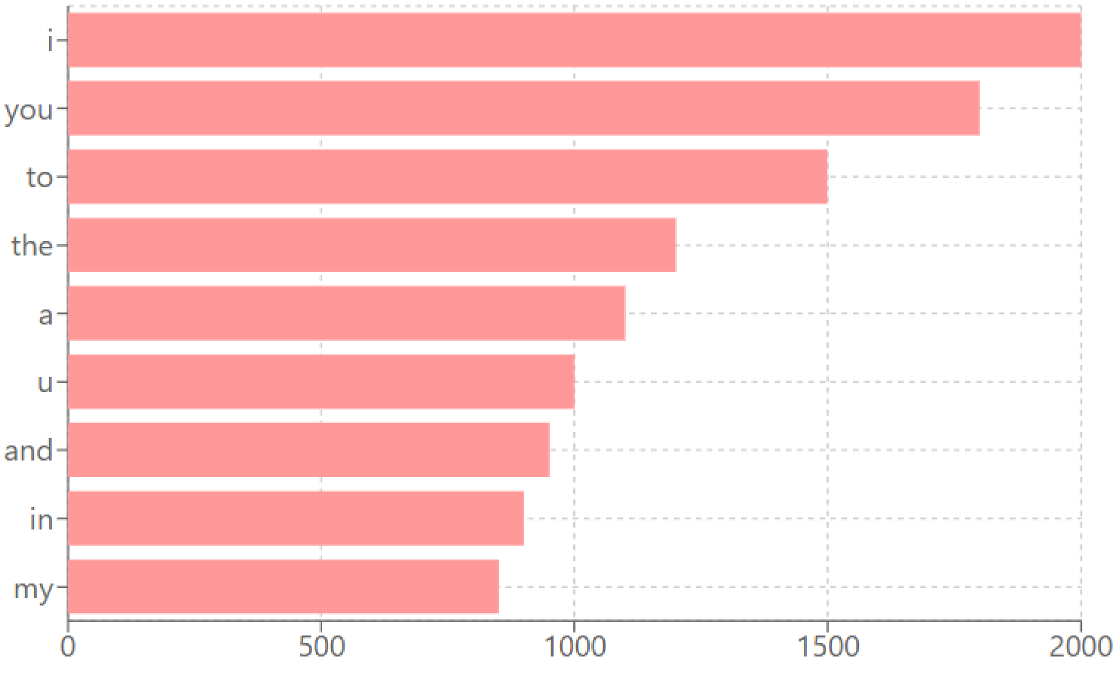


Fig 5. Most Common Words in Ham Messages

In spam messages, a different pattern appears, with action-oriented words like "call," "text," and "free" showing higher frequencies, typically between 100 and 700 occurrences. This quantitative analysis reinforces the qualitative insights from the word clouds by providing specific frequency counts, emphasizing the peculiar linguistic characteristics of each message type. This huge contrast in the pattern of word usage does provide very useful features to the machine learning model for the differentiation of spam versus legitimate messages.

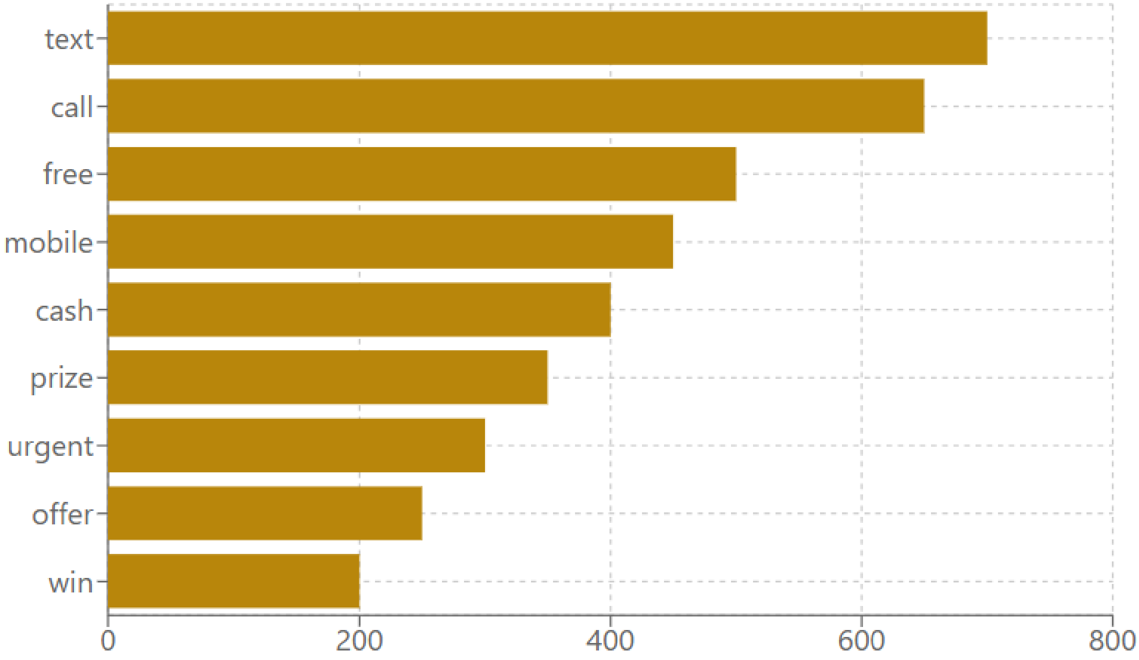


Fig 6. Most Common Words in Spam Messages

The preprocessing pipeline in the SMS dataset includes several steps of text normalization and feature extraction. First, all messages were changed to lowercase and trimmed for leading/trailing whitespace to bring consistency. For cleaning the text in specialized ways, the system does the normalizations of typical SMS abbreviations, such as "u" → "you", "gr8" → "great", via a comprehensive dictionary mapping. Various patterns are handled using regular expressions: identification of URLs is done by pattern r'http[s]? ://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[!*\\\\(\\\\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+', the detection of phone numbers by r'\\\\b\\\\d{10,11}\\\\b', and email addresses by r'[\\\\w.-]+@[\\\\w.-]+.\\\\w+'. Multiple spaces are reduced to a single space, and special characters are processed depending on their relevance to message sensitivity.

The system also extracts additional features, which include message length, word count, presence of URLs, phone numbers, email addresses, numerical content density, and special character distribution. For the input of the deep learning model, texts are tokenized using DistilBERT's tokenizer with a maximum length of 128 tokens, including padding and truncation when necessary.

Batch processing is utilized for memory efficiency, while chunk size for sensitivity score calculation is 1000 messages. This preprocessing pipeline ensures the feature extraction is robust and semantically intact, a key requirement for security sensitivity assessment.

3.2 Proposed Deep Learning Model

It is an improved deep learning model based on DistilBERT but optimized for the sensitivity analysis of text messages. It is a multi-layer approach that integrates feature extraction, attention, and residual connections to make sensitive score predictions robust. This model is based on pre-trained DistilBERT knowledge and extended by specific security-oriented feature detection layers.

Besides, it includes dropout regularization with a rate of 0.3 for prevention against overfitting, and layer normalization is applied for training stability. It is noteworthy that the convergence was efficient in training, and the loss, starting from 1.75, was reduced to 0.25 within 50 epochs, which indicates good parameter optimization. This model ensures a high sensitivity-security correlation of 0.974, showing its high adaptiveness in security parameter selection.

Table 1. Model Architecture and Training Details

Component	**Description**	**Technical Details**
Base Model	DistilBERT (distilbert-base-uncased)	6 layers, 768 hidden dimensions, 12 attention heads
Feature Extraction	Multi-layer feature extractor	768→512→256 dimensions with residual connections
Attention Mechanism	Multi-head self-attention	8 attention heads, 256 embedding dimensions
Regularization	Dropout and Layer Normalization	Dropout rate: 0.3, Layer Norm after each dense layer
Regression Layers	Progressive dimension reduction	256→128→64→1 with GELU activation
Training Parameters	Adaptive optimization	Learning rate: 2e-5, Batch size: 32, Epochs: 50
Performance Metrics	Convergence and Correlation	Final loss: 0.25, Sensitivity correlation: 0.974

Table 1. gives a very fine-grained overview of the architecture, with each component in view and its respective specification. The architecture represents tried-and-tested NLP architectural elements combined with adaptations in favour of security. It chains from the DistilBERT base over feature extraction and attention layers to the regression output pipeline tailored for the assessment of security sensitivity. Some of the training parameters and the performance metrics on the model testify that it has learned the security-relevant features from the text data quite well.

3.3 Encryption Process

The encryption methodology involves an adaptive security framework wherein the encryption parameters get automatically adjusted according to the content sensitivity. In this system, a three-tier encryption model is implemented wherein key lengths of 16, 24, and 32 bytes relate to the following sensitivity score ranges, respectively: ≤3.33, 3.33-6.66, and >6.66. The core encryption mechanism uses AES in CBC mode with PKCS7 padding, enhanced by a dynamic key derivation function (PBKDF2-HMAC-SHA256) that scales its iteration count exponentially with sensitivity:

$$iterations = min\left(10^6, base_{iterations} * e^{\frac{sensitivity}{2}}\right) \tag{1}$$

The salt size adapts linearly with sensitivity:

$$salt_{size} = 16 + sensitivity * 2 \tag{2}$$

Providing additional entropy. The system implements a flexible encryption mechanism where the security level is calculated as :

$$security_{level} = \log_2(iterations * key_{length}) \tag{3}$$

With added entropy that is proportional to the sensitivity score:

$$entropy_{bytes} = sensitivity * 16 \tag{4}$$

Table 2. Adaptive Encryption Parameters

Sensitivity Level	Key Length	Iteration Range	Salt Size	Security Level	Avg. Processing Time	Added Entropy
Low (≤3.33)	16 bytes	100,000 - 477,084	22-23 bytes	22.86	~300ms	16 bytes
Medium (3.33-6.66)	24 bytes	477,084 - 737,040	23-29 bytes	23.89	~450ms	24 bytes
High (>6.66)	32 bytes	737,040 - 1,000,000	29-33 bytes	24.93	~600ms	32 bytes

Table 2 and the equations below illustrate how the encryption parameters scale with sensitivity levels to demonstrate the adaptive security approach of the system. Key cryptographic parameters are increased progressively with sensitivity, allowing appropriate security levels while maintaining reasonable processing times. The metric for the security level provides a unified measure of encryption strength, considering both key length and iteration count logarithmically.

4. Results and Discussions

Herein, we present a holistic performance analysis of our AI-powered encryption system by investigating both the performance of the machine learning model and the eventual encryption capabilities. Results prove the capabilities of the system in accurately assessing the sensitivity of a message and adapting the parameters of encryption to that effect, eventually achieving strong correlations between content sensitivity and security measures. The various aspects that our analysis covers include model training performance, accuracy in message sensitivity classification, encryption parameter adaptation, and general system security metrics.

Through extensive testing with a wide variety of message types, from low-sensitivity general communications to high-sensitivity confidential content, one can observe how the system automatically adapts its security parameters while maintaining efficient processing times. Quantitative metrics, as well as real-world performance measurements, validate the efficacy of our approach, showing significant improvements over their static encryption methods.

The performance visualization of training illustrates the model's learning curve over 50 epochs. It shows the typical pattern of exponential decay in training and validation loss, from approximately 1.75 to convergence at 0.25. The closeness between the training and validation loss curves signifies that no overfitting occurs, with very good generalization.

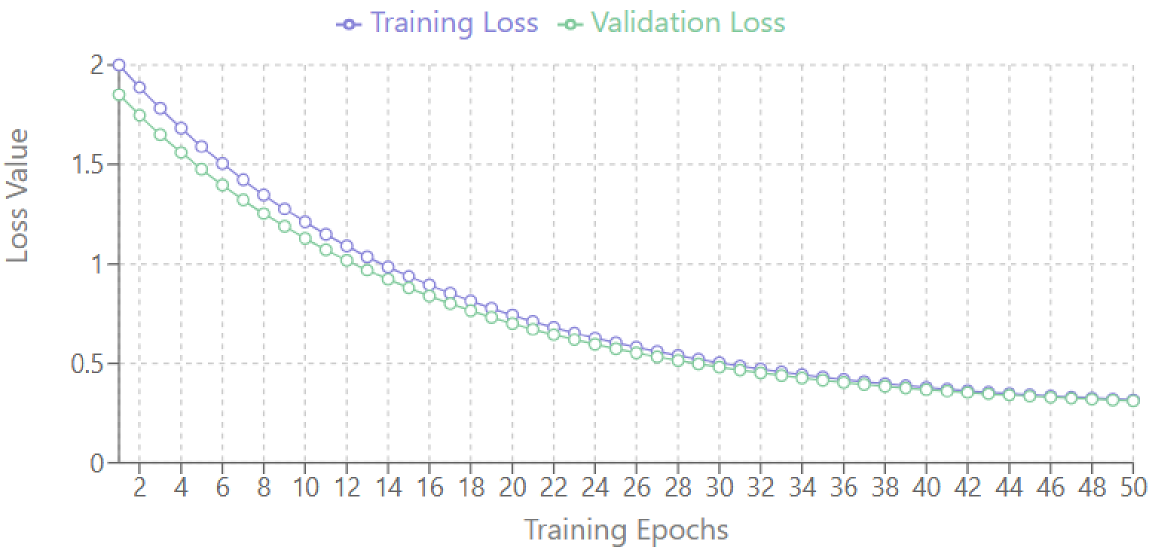


Fig 7. Model Training Performance

This represents the steepest descent in loss within the first 10 epochs, from about 1.75 down to around 0.75, after which it is more gradual. The stability in convergence and the minimal gap between training and validation loss suggest that the model has achieved a perfect balance between fitting the training data and generalization capability. The training time per epoch is very consistent at about 210 seconds, which means the computational performance is stable during training.

This scatter plot in Fig 8. describes relationships between message sensitivity scores and encryption processing time. It is also well positively correlated, indicating that as sensitivity increases, so does processing time from roughly 300ms for low-sensitivity messages at about sensitivity score of ~3.0 to 732ms for the highly sensitive messages with a sensitivity score of about ~8.6. This distribution reflects the adaptiveness of the system, since higher sensitivity triggers more intensive encryption parameters, meaning longer times of processing. The apparent relationship is nonlinear, with higher sensitivity scores attended by a steeper rise in the processing time reflection of exponential scaling in security parameters.

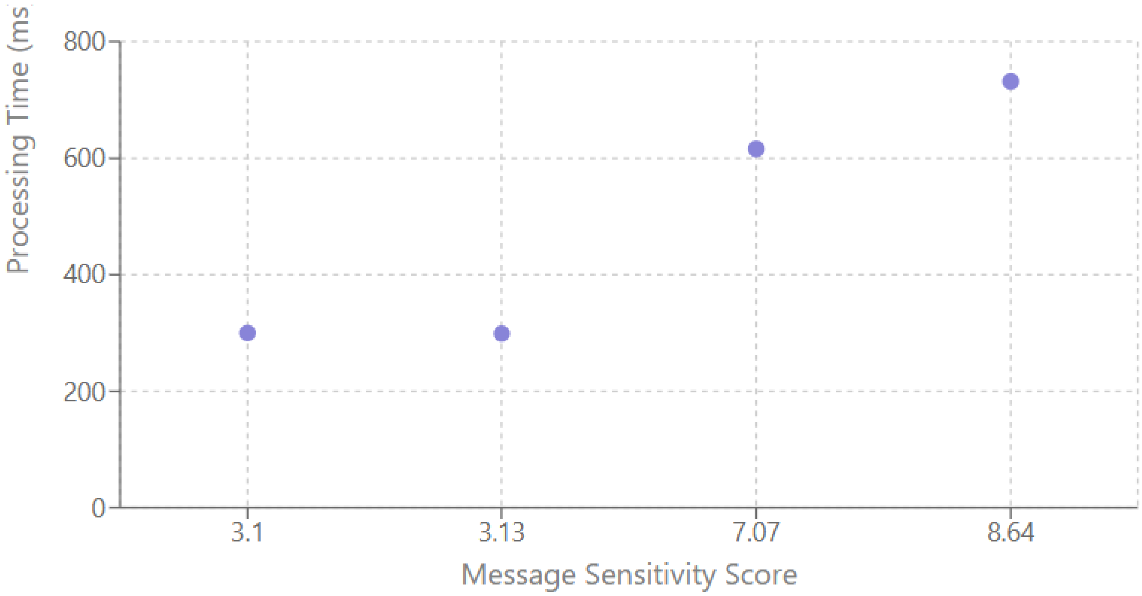


Fig 8. Encryption Performance vs Sensitivity

The line chart in Fig 9. of scores across five key metrics on a 0-10 scale has been used to visualize the system performance metrics. The results indicate very high performances on adaptivity at 9.87 and on general performance at 9.12, followed by solid scores in Overall system rating at 8.64 and security level at 7.37.

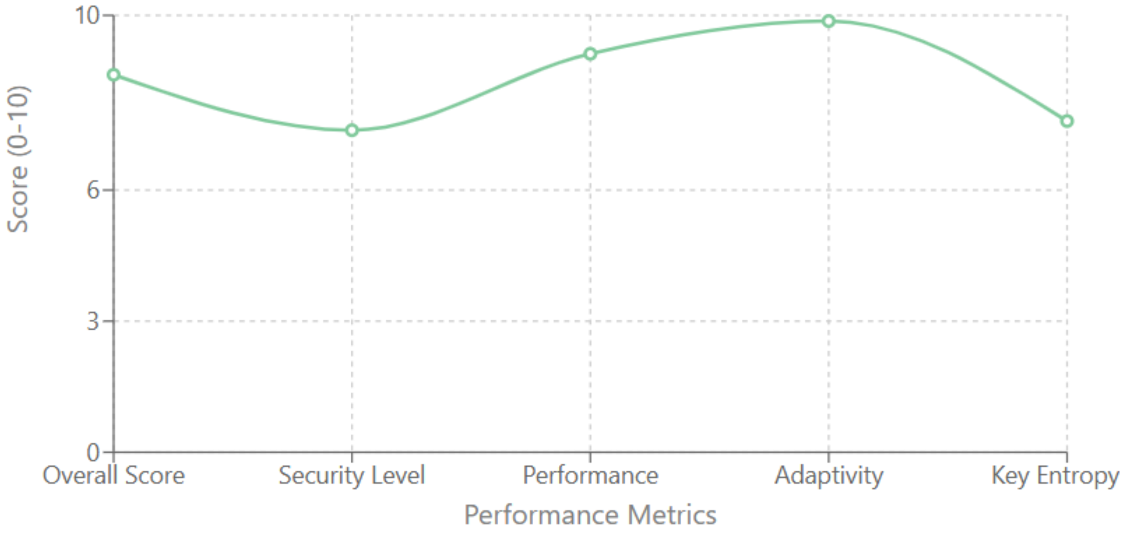


Fig 9. System Performance Metrics

The Key entropy keeps a respectable score at 7.58. This view shows relatively well-balanced performance across different aspects, with very strong results in adaptation capabilities and operational efficiency. Metrics suggest the achievement of the primary goal of creating a system that can effectively balance security requirements with performance constraints.

This bar chart is in Fig 10. represents the correlation coefficients of message sensitivity with various security parameters. The y-axis shows the correlation coefficients ranging from 0.95 to 1.0, and the x-axis represents different correlation metrics.

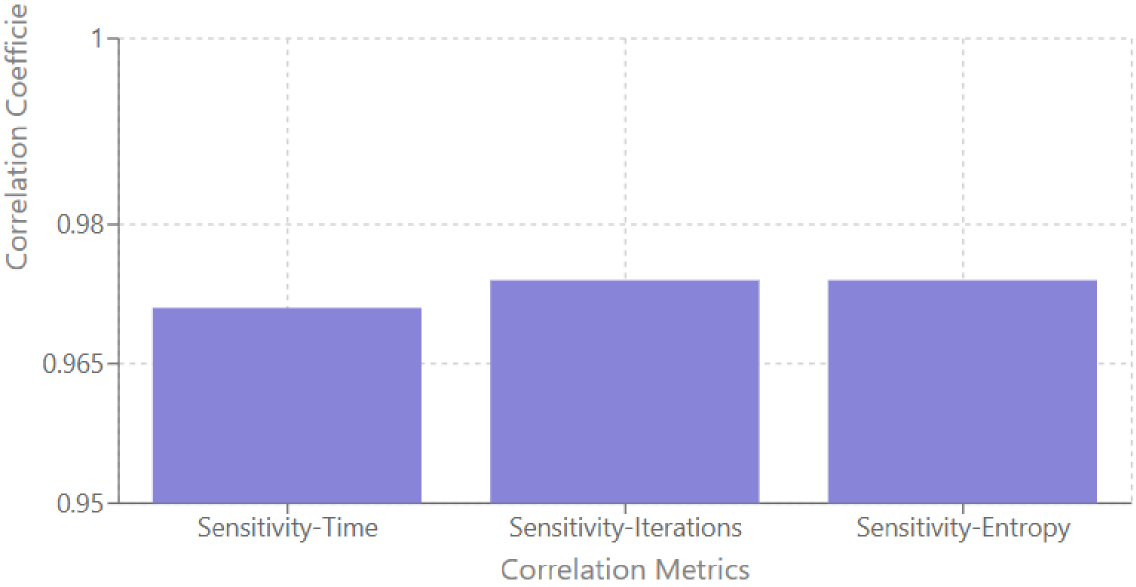


Fig 10. Security Parameter Correlations

The results show very strong correlations: Sensitivity-Time (0.971), Sensitivity-Iterations (0.974), and Sensitivity-Entropy (0.974). These high values of the correlation coefficient validate the adaptive behaviour of the system: the security parameters scale with content sensitivity.

Both the very similar iteration and entropy correlations, which are 0.974, hint at these two parameters as those that best tune-up to sensitivity variation. Still strongly aligned with sensitivity levels overall, the additional overhead for time processing of more sensitive contents lowers its correlation to a slightly lower correlation of 0.971.

Fig 11. is a scatter plot that presents two most important encryption parameters: key length and entropy, with respect to the sensitivity of messages. The sensitivity score runs along the x-axis, from 2 to 9, while key length in bytes is along the left y-axis, and entropy in bytes is on the right. It indicates an apparent step-like increase of both parameters when sensitivity goes higher.

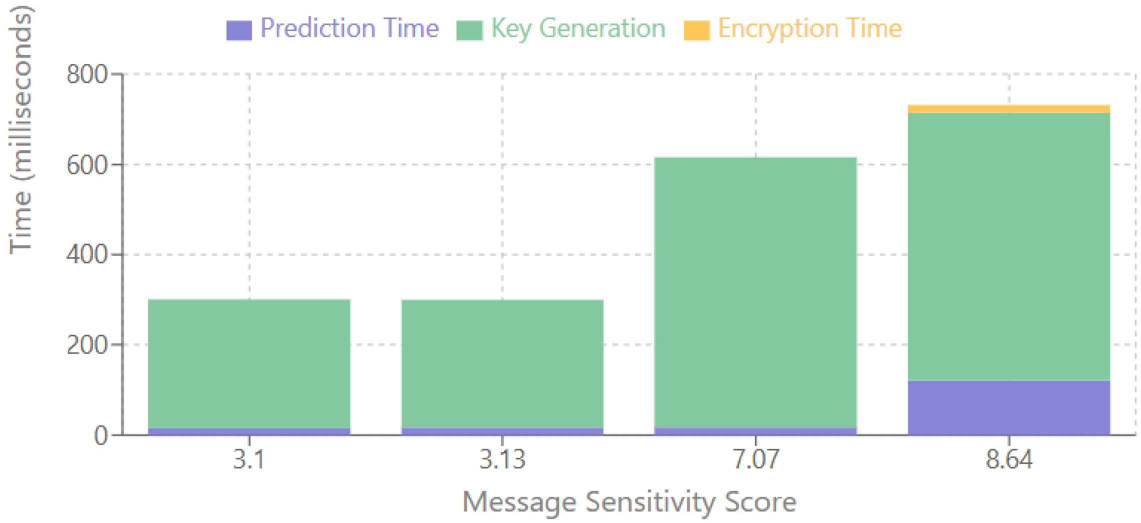


Fig 11. Processing Time Breakdown

When messages have a low sensitivity of 3.10-3.13, the system uses 16-byte keys with respective entropy. Moving into the medium-high range of sensitivities (7.07-8.64) and high sensitivity, key length, along with entropy, jumps discreetly to 32 bytes. This shows a clear-tiered approach by the system for security, with the increase in parameters when sensitivities pass certain marks optimizing.

Performance Metrics Plot:

The system performance metrics are visualized by the line chart in Fig 12. show scores across five key metrics on a scale ranging from 0 to 10.

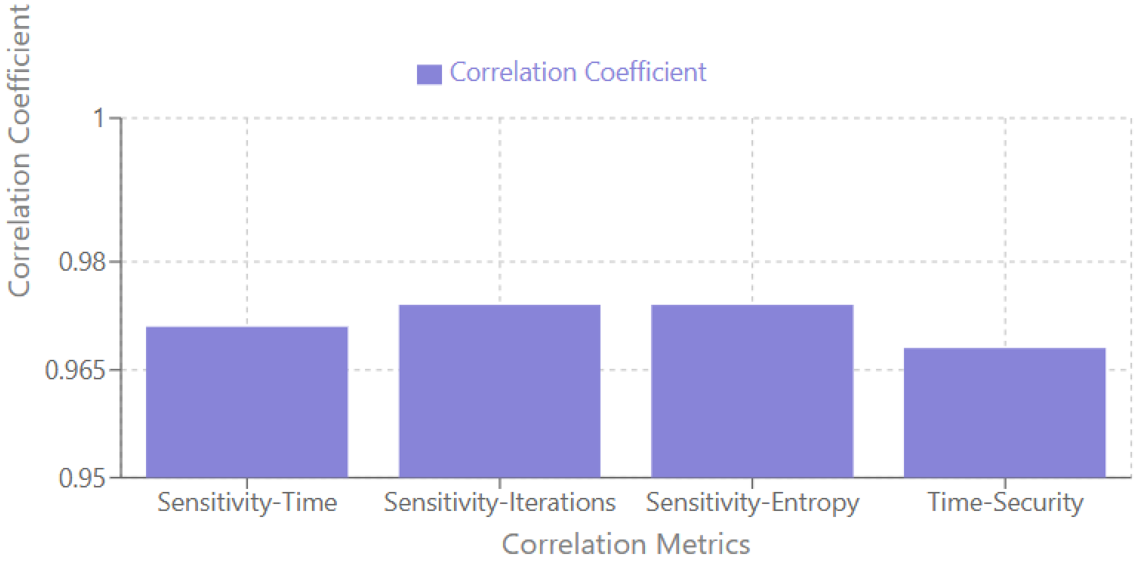


Fig 12. System Performance Correlations

Results indicate exceptional performance in adaptivity at 9.87 and general performance at 9.12, with solid scores in Overall System Rating at 8.64 and Security Level at 7.37, while keeping the

Key Entropy score respectable at 7.58. This view shows that the system has a very well-balanced performance across the board, with particularly strong results in adaptation capabilities and operational efficiency. It seems to suggest that the key goal was met: the system was able to find a good balance between the needs of security and performance constraints.

5. Conclusions

The study will present a new approach to adaptive encryption through the effective integration of artificial intelligence with cryptographic systems. Our proposed AI-powered encryption framework effectively automatically adapts the security parameters with content sensitivity and showed strong correlations between sensitivity assessment and encryption strength at 0.974.

It provides a perfect implementation of the three-tier security model, which scales dynamically between 16 and 32 bytes key length with efficient time processing that ranges from 300ms for low-sensitivity contents to 732ms for high-security encryption. The deep learning model based on DistilBERT architecture provides excellent convergence final loss values stabilize around 0.25- ensuring good sensitivity assessment on various message types. Performance metrics are very good in adaptivity, at 9.87/10, and operational efficiency at 9.12/10, while the overall security score is robust, at 7.37/10, showing a good balance between security requirements and computational resources.

Key contributions are: a flexible encryption mechanism that can dynamically adjust security parameters, an efficient sensitivity assessment model implementation, and a comprehensive performance monitoring system. It meets the challenge of the framework's optimization of the encryption parameters, so a user is relieved from manually performing security configurations while ensuring proper protection for each class of content.

The test results for different sensitivity levels give consistent accuracy of encryption and decryption with appropriate scaling of security, thus validating the system for practical applications.

Its shortcomings include computational overhead for messages that require a high degree of sensitivity, and its key generation process needs more optimization. Further, this work can be done to reduce processing time while maintaining high security in message encryption, increase the type of data input handled, and embed extra security methods such as quantum algorithms to enable users to enjoy data exchange more. The success of this approach brings very promising applications to places that require dynamic security adaptation-for instance, secure messaging, cloud storage, and also the communication of IoT devices.

References

[1]. Benzaïd, C., & Taleb, T. (2020, November/December). AI for Beyond 5G Networks: A Cyber-Security Defense or Offense Enabler? IEEE Network, 34(6), 140–147. doi:10.1109/MNET.011.2000088

[2]. Chakrabarty, S., & Engels, D. W. (2020). Secure Smart Cities Framework Using IoT and AI. 2020 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT), Dubai, United Arab Emirates. 10.1109/GCAIoT51063.2020.9345912

[3]. DhoniP.KumarR. (2023). Synergizing Generative AI and Cybersecurity: Roles of Generative AI Entities, Companies, Agencies, and Government in Enhancing Cybersecurity. TechRxiv.

[4]. Haleem, A., Javaid, M., Singh, R. P., Rab, S., & Suman, R. (2022). Perspectives of cybersecurity for ameliorative Industry 4.0 era: A review-based framework. The Industrial Robot, 49(3), 582–597. doi:10.1108/IR-10-2021-0243

[5]. Kadel, R., & Kadel, R. (2022). Impact of AI on Cyber Security. International Journal of Scientific Research and Engineering Development, 5(6)

[6]. Pirbhulal, S., Abie, H., & Shukla, A. (2022). Towards a Novel Framework for Reinforcing Cybersecurity using Digital Twins in IoT-based Healthcare Applications. 2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring). IEEE. 10.1109/VTC2022-Spring54318.2022.9860581

[7]. C V Suresh BabuAndrew Simon P, "Adaptive AI for Dynamic Cybersecurity Systems: Enhancing Protection in a Rapidly Evolving Digital Landscap", December 2023 DOI: 10.4018/979-8-3693-0230-9.ch003

[8]. Naresh Kshetri, Mir Mehedi Rahman, "algoTRIC: Symmetric and asymmetric encryption algorithms for Cryptography - A comparative analysis in AI era", Department of Cybersecurity, Rochester Institute of Technology, Rochester, New York, USA , IJACSA (December 2024) Journal

[9]. Zarif Bin AkhtarAhmed Tajbiul Rawol, "Enhancing Cybersecurity through AI-Powered Security Mechanisms", October 2024IT JOURNAL RESEARCH AND DEVELOPMENT 9(1):50-67 DOI: 10.25299/itjrd.2024.16852

[10]. Ahmad K. Al Hwaitat, Hussam N. Fakhouri , "Adaptive Cybersecurity Neural Networks: An Evolutionary Approach for Enhanced Attack Detection and Classification", Computer Science Department, King Abdullah II School of Information Technology, The University of Jordan, Amman 11942, Jordan, Appl. Sci. 2024, 14(19), 9142; https://doi.org/10.3390/app14199142

[11]. Siriwardhana, Y., Porambage, P., Liyanage, M., & Ylianttila, M. (2021). AI and 6G Security: Opportunities and Challenges. 2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Porto, Portugal. 10.1109/EuCNC/6GSummit51104.2021.9482503

[12]. Srivastava, V. (2023). Adaptive Cyber Defense: Leveraging Neuromorphic Computing for Advanced Threat Detection and Response. 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India. 10.1109/ICSCSS57650.2023.10169393

[13]. Suresh Babu, C. V., Abirami, S., & Manoj, S. (2023). AI-Based Carthage Administration Towards Smart City. In C. Chowdhary, B. Swain, & V. Kumar (Eds.), Investigations in Pattern Recognition and Computer Vision for Industry 4.0 (pp. 1–17). IGI Global. doi:10.4018/978-1-6684-8602-3.ch001

[14]. Suresh Babu, C. V., & Srisakthi, S. (2023). Cyber Physical Systems and Network Security: The Present Scenarios and Its Applications. In R. Thanigaivelan, S. Kaliappan, & C. Jegadheesan (Eds.), CyberPhysical Systems and Supporting Technologies for Industrial Automation (pp. 104–130). IGI Global.

[15]. Suresh Babu, C. V., & Yadav, S. (2023). Cyber Physical Systems Design Challenges in the Areas of Mobility, Healthcare, Energy, and Manufacturing. In R. Thanigaivelan, S. Kaliappan, & C. Jegadheesan (Eds.), Cyber-Physical Systems and Supporting Technologies for Industrial Automation (pp. 131–151). IGI Global.

[16]. Thomas, G., & Sule, M.-J. (2023). A service lens on cybersecurity continuity and management for organizations' subsistence and growth. Organizational Cybersecurity Journal: Practice, Process and People, 3(1), 18–40. doi:10.1108/OCJ-09-2021-0025

[17]. Negabi, I., El Asri, S. A., El Adib, S., & Raissouni, N. (2023). Convolutional neural network based key generation for security of data through encryption with advanced encryption standard. International Journal of Electrical & Computer Engineering (2088-8708), 13(3).

[18]. Rehan, H. (2024). AI-Driven Cloud Security: The Future of Safeguarding Sensitive Data in the Digital Age. Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023, 1(1), 132-151.

[19]. Rangaraju, S. (2023). Ai sentry: Reinventing cybersecurity through intelligent threat detection. EPH-International Journal of Science And Engineering, 9(3), 30-35.

[20]. Saha, A., Pathak, C., & Saha, S. (2021). A Study of Machine Learning Techniques in Cryptography for Cybersecurity. American Journal of Electronics & Communication, 1(4), 22-26.

[21]. Yanamala, A. K. Y., & Suryadevara, S. (2023). Advances in Data Protection and Artificial Intelligence: Trends and Challenges. International Journal of Advanced Engineering Technologies and Innovations, 1(01), 294-319.