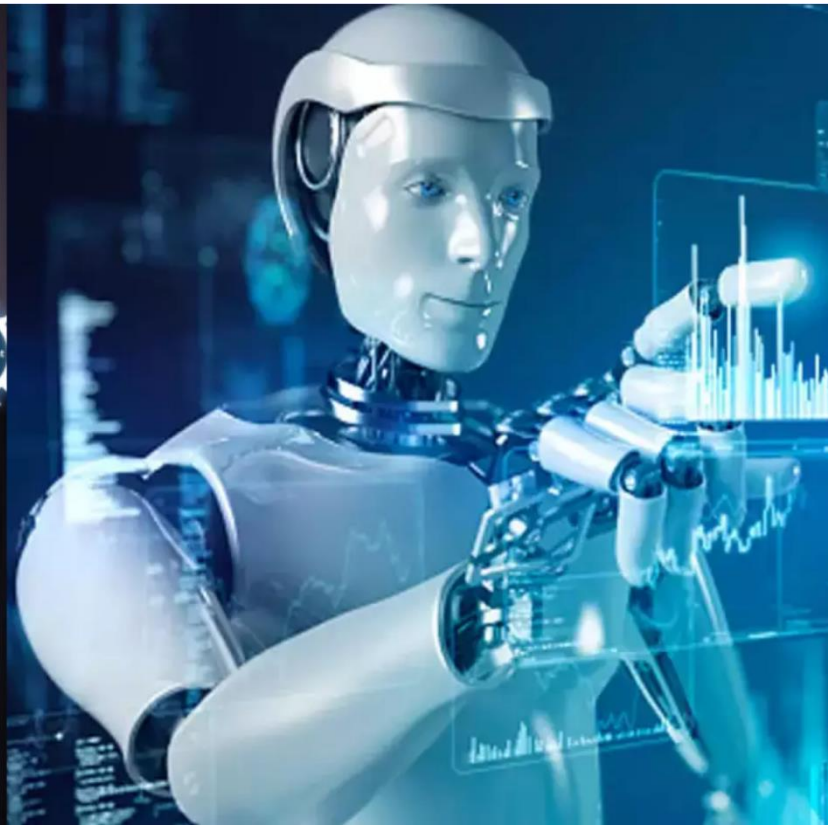
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 5, May 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

Cryptographic Algorithm Detection from Dataset using AI/ML Techniques

Bhavya Shree T, Rohana T, Sirisha S, Sneha V

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickaballapura, Karnataka, India

Prof. Anand Tilagul

Assistant Professor, Department of Information Science and Engineering, S J C Institute of Technology, Chickaballapur, Karnataka, India

ABSTRACT: This paper presents an AI/ML-based approach for detecting cryptographic algorithms from datasets. By extracting statistical and entropy-based features, machine learning models classify encryption types with high accuracy. The proposed method enhances automated cryptographic analysis, supporting cybersecurity tasks such as malware detection, secure protocol verification, and cryptographic compliance auditing.

KEYWORDS: Machine Learning, Cryptographic Algorithm, Classification, Random Forest, XGBoost, AES, DES, RSA, Blowfish, Ciphertext.

I. INTRODUCTION

With the rapid growth of digital communication, protecting data through encryption has become essential across many industries, including finance, healthcare, and government. Cryptographic algorithms are used to secure this data, but identifying which algorithm has been applied to encrypted data (ciphertext) remains a major challenge—especially when the encryption key is not available. Traditional methods of identifying cryptographic algorithms often require access to the encryption key or detailed knowledge of the algorithm, which is not always practical. To address this, we propose a machine learning-based approach that can identify cryptographic algorithms by analyzing features of the ciphertext alone. In this study, we extract key statistical features from ciphertext, such as byte frequency, Shannon entropy, block size, key length, ciphertext length, n-gram frequency, and statistical moments like mean and variance. These features are used to train machine learning models—specifically Random Forest and XGBoost—to classify different cryptographic algorithms, including AES, DES, RSA, Blowfish, RC4, Twofish, and Triple DES (3DES).This approach does not require access to encryption keys and offers an efficient, accurate way to identify algorithms. It can be useful in areas such as cybersecurity analysis, digital forensics, and academic research, where understanding the type of encryption used is important but direct access to keys is not possible.

II. PROBLEM STATEMENT

The growing use of cryptographic algorithms for securing digital communications necessitates the ability to identify and distinguish between different algorithms based on their ciphertext characteristics. Traditional methods of algorithm detection often rely on access to key information or detailed algorithm-specific knowledge. However, in many cases, such access is unavailable, and there is a need for alternative techniques to classify cryptographic algorithms. This project addresses the challenge of identifying cryptographic algorithms by analyzing the ciphertext alone. By leveraging machine learning techniques, it aims to classify various cryptographic algorithms, such as AES, DES, RSA, Blowfish, RC4, Twofish, and Triple DES, based on extracted features like byte frequency, entropy, block size, key length, ciphertext length, n-gram frequency, and statistical moments. This approach allows for accurate classification without requiring the decryption key, making it a promising solution for situations where cryptographic analysis is crucial, but key access is not possible.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

III. LITERATURE REVIEW

1. **Singh, J., & Yadav, V. K. (2017)** – "A survey of cryptographic algorithm classification using machine learning".
   Outcome: This paper surveys various machine learning techniques used for classifying cryptographic algorithms based on ciphertext features like entropy and byte frequency.
   Disadvantage: It lacks practical implementation and comparative performance metrics.
2. **Jang, Y., & Lee, M. (2019)** – "Cryptanalysis of block ciphers using machine learning".
   Outcome: The authors applied neural networks to automate the process of cipher pattern recognition and key recovery.
   Disadvantage: The approach requires large datasets and faces difficulties with model generalization across different ciphers.
3. **Herscovici, D. (2018)** – "Machine learning for identifying encryption algorithms".
   Outcome: The study successfully classified encryption algorithms using decision trees and support vector machines on extracted ciphertext features.
   Disadvantage: There is a trade-off between model complexity and accuracy, impacting scalability.
4. **Chin, A., & Ghosh, A. (2020)** – "Feature extraction techniques for cryptographic algorithm identification".
   Outcome: This research highlighted effective statistical features like n-gram frequency and entropy that significantly improve classification accuracy.
   Disadvantage: The feature extraction process can be computationally expensive and requires manual tuning.
5. **Srinivasan, S., & Ramachandran, M. (2021)** – "Machine learning techniques for the identification of cryptographic algorithms".
   Outcome: Demonstrated how Random Forest and XGBoost models can classify algorithms like AES and RSA with high accuracy.
   Disadvantage: Focused mainly on traditional algorithms and synthetic datasets, limiting real-world applicability.
6. **Carvalho, F., & Lima, A. (2020)** – "Evaluation of machine learning techniques for cryptographic algorithm classification".
   Outcome: Compared multiple ML models and showed that ensemble models perform best for algorithm detection.
   Disadvantage: The study didn't explore model behavior under adversarial or noisy data conditions.
7. **Alharthi, A. M., & Alqahtani, A. S. (2020)** – "A comparative study of cryptographic algorithm identification using statistical features".
   Outcome: Identified that combining entropy with byte-level statistics enhances classification performance.
   Disadvantage: The results are based on limited algorithm types and do not generalize well to hybrid encryption schemes.
8. **Zhu, Z., & Yang, J. (2019)** – "Identification of encryption algorithms based on machine learning classifiers: A survey".
   Outcome: Reviewed ML classifiers like KNN, Naive Bayes, and neural networks for encryption detection.
   Disadvantage: Provided minimal insight into feature selection or dataset construction.
9. **Srinivasan, R., & Deepika, K. (2017)** – "Cryptographic algorithm classification using machine learning methods".
   Outcome: Focused on building lightweight models for fast classification of symmetric encryption algorithms.
   Disadvantage: Lacks robustness for asymmetric and modern post-quantum algorithms.
10. **Kumar, N., & Kaur, P. (2018)** – "A machine learning approach for the classification of cryptographic algorithms".
    Outcome: Proposed a model using decision trees and random forests for real-time algorithm classification.
    Disadvantage: Did not evaluate the performance under varying input lengths or encryption modes.

IV. DESIGN AND IMPLEMENTATION

The proposed system for cryptographic algorithm detection is built using a machine learning-based pipeline. It is structured into four main components: Data Collection, Feature Extraction, Model Training, and Algorithm Classification.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

1. Data Collection
In this stage, a custom dataset is created by encrypting sample plaintext files using various symmetric cryptographic algorithms such as AES, DES, Blowfish, and RC4. Each algorithm is applied using different keys and modes to ensure a diverse and representative dataset. This ciphertext data, along with corresponding algorithm labels, forms the basis for training and testing.

2. Feature Extraction
Once the ciphertext is generated, several important features are extracted to serve as input for the machine learning models. These include:

Shannon Entropy: Measures the randomness of the ciphertext.
Byte Frequency Distribution: Analyzes how frequently each byte (0–255) appears.
n-Gram Statistics: Captures byte pattern sequences that can be unique to specific encryption algorithms.
Block Size & Data Length: Additional metadata that may help in distinguishing between block and stream ciphers.
These features help convert raw ciphertext into numerical data that ML models can understand and learn from.

3. Model Training
After feature extraction, various machine learning models are trained using the labeled dataset. Algorithms such as Random Forest, Support Vector Machines (SVM), and K-Nearest Neighbors (KNN) are tested for their performance in classifying the encryption type. During training, cross-validation techniques are used to prevent overfitting and ensure generalization.

4. Algorithm Classification
In the final stage, the trained model is used to classify unseen ciphertext samples. The system analyzes the features of the input ciphertext and predicts the cryptographic algorithm used. The output is evaluated using metrics like accuracy, precision, recall, and F1-score to assess model performance.

This design allows the system to detect algorithms without requiring access to encryption keys and provides a fast, scalable solution for automated cryptographic analysis.

Activity Diagram

The Activity Diagram below illustrates the stepwise flow of user interactions and system processes in deepfake detection:

Stepwise Explanation:

1. Start
The process begins when the user opens or launches the application. This initialization step ensures that the necessary background services, interfaces, and resources are ready for interaction.

2. User Registers/Login
The system prompts the user to register (if new) or log in (if existing).
Registration involves storing user credentials securely using hashing techniques like bcrypt or SHA-256.
Login verifies user credentials and grants access to the application's core functionality.
Security protocols such as Two-Factor Authentication (2FA) may be integrated for additional protection.

3. Input Data – Enter Ciphertext Features
Once authenticated, the user provides input data—ciphertext features.
These features are derived from encrypted text and may include:
Byte distribution,Frequency of characters,Entropy measures,Bit patterns
The input can be raw ciphertext or structured features derived from it.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

4. Process Data – Preprocess & Extract Features
This stage transforms raw input into a format suitable for machine learning models:
Preprocessing: Removes noise, handles missing values, and scales or encodes data.
Feature Extraction: Computes statistical or structural properties from the ciphertext, e.g.:
Entropy,N-gram frequency,Character class distribution
This ensures that the model can generalize well during training and inference.

5. Train Model
A machine learning model (e.g., Decision Tree, Random Forest, SVM, or Neural Network) is trained on labeled data.
Input: Preprocessed feature vectors from known cryptographic algorithms.
Output: A trained classifier capable of recognizing patterns in ciphertext features.
Training can be done offline or periodically with updated datasets to improve accuracy.

6. Predict Algorithm
The trained model is used to analyze the new input data and predict the encryption algorithm used (e.g., AES, DES, RSA).
This is a classification task, where the model assigns the input to one of several known algorithms.

7. Display Result
The prediction result is presented to the user.
This includes:
The predicted cryptographic algorithm ,Possibly a confidence score or probability distribution
Time taken or any relevant metrics
The result helps in forensic analysis, algorithm classification, or security auditing.

8. End
The workflow concludes. The user may:
Restart the process with new data
Save results
Log out or exit the application

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

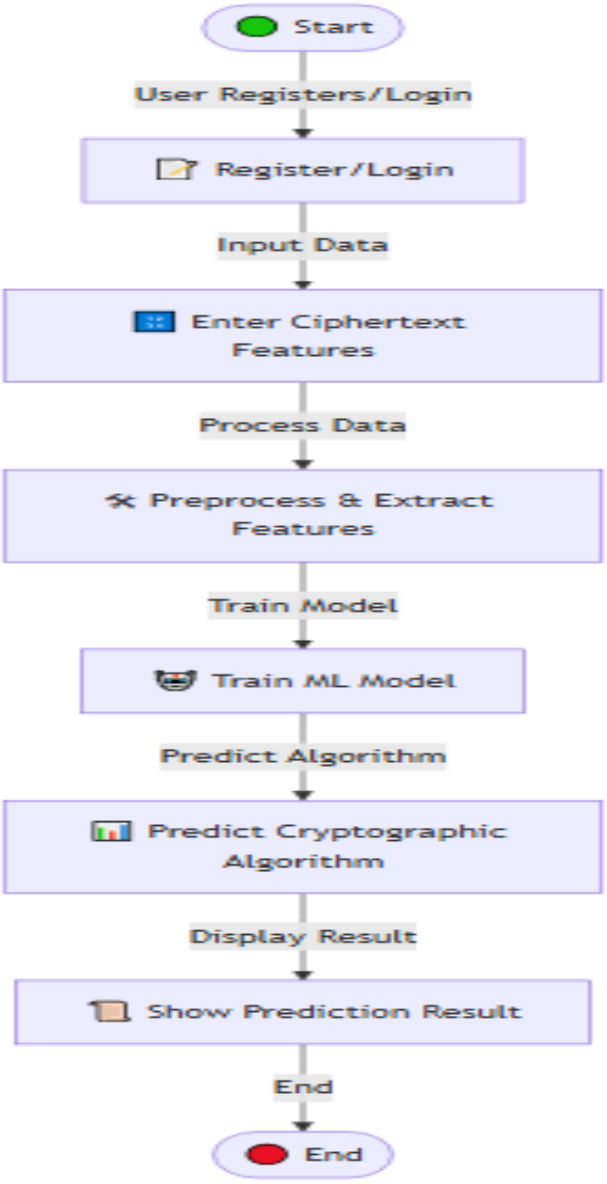


Fig. 1. Activity Diagram of Cryptographic algorithm detection

The activity diagram provides a visual representation of the step-by-step workflow in the cryptographic algorithm detection system.

Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis.

Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

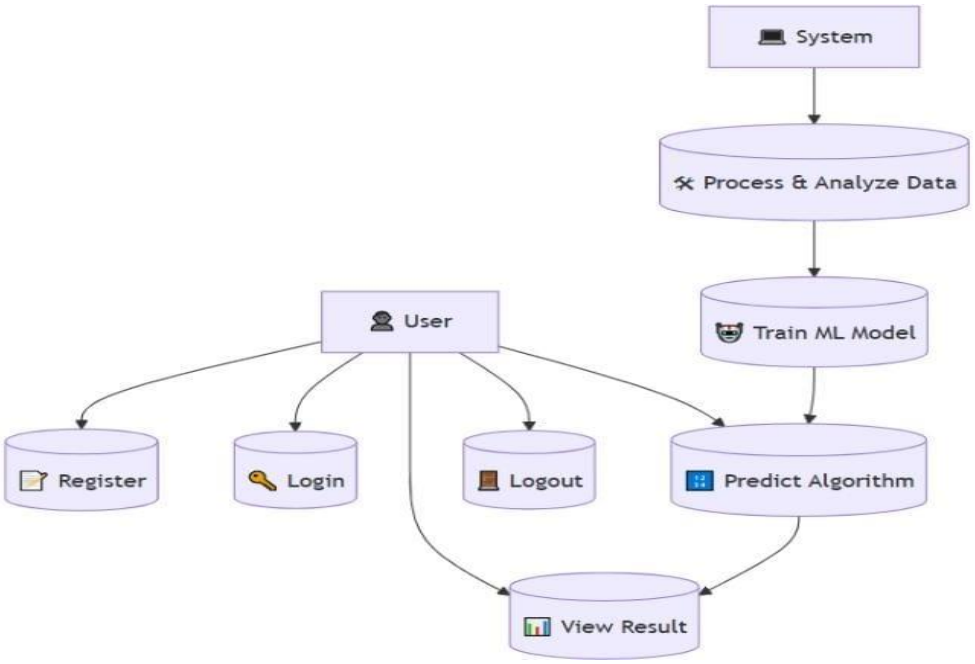


Fig. 2. Use Case Diagram for Cryptographic algorithm detection

The system allows users to register, log in securely, and log out after use. Once logged in, users can input ciphertext features to predict the encryption algorithm. The system processes these inputs and uses a trained machine learning model to generate predictions, such as "AES-256 with 92% confidence." Users can then view the result along with confidence levels. In the background, the system handles data cleaning, feature extraction, and model training to ensure accurate and efficient predictions without needing encryption keys.

V. CONCLUSION

In conclusion, this project demonstrates the effectiveness of machine learning algorithms in identifying cryptographic algorithms from ciphertext analysis. By focusing on statistical features extracted from the ciphertext, the proposed system eliminates the need for encryption key access, thus offering a scalable and efficient solution for cryptographic analysis. The use of Random Forest and XGBoost models has shown promising results in classifying a range of cryptographic algorithms with high accuracy, making it a valuable tool for cybersecurity, cryptanalysis, and even educational purposes. Traditional methods of algorithm identification, which rely on manual inspection or access to encryption keys, are time-consuming and prone to human error, whereas the machine learning-based approach provides a more reliable and automated alternative. This system has significant potential in real-world scenarios where decryption keys are inaccessible, contributing to more robust cybersecurity practices and providing insight into the strengths and weaknesses of existing encryption methods.

REFERENCES

[1] Singh, J., & Yadav, V. K. (2017). A survey of cryptographic algorithm classification using machine learning. International Journal of Computer Applications, 165(6), 1–5.
[2] Jang, Y., & Lee, M. (2019). Cryptanalysis of block ciphers using machine learning. International Journal of Computer Science and Network Security, 19(8), 87–92.
[3] Herscovici, D. (2018). Machine learning for identifying encryption algorithms. Procedia Computer Science, 141, 204–211.
[4] Chin, A., & Ghosh, A. (2020). Feature extraction techniques for cryptographic algorithm identification. Advances in Intelligent Systems and Computing, 962, 315–323.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405318|

[5] Srinivasan, S., & Ramachandran, M. (2021). Machine learning techniques for the identification of cryptographic algorithms. IEEE Xplore, 12(4), 32–40.
[6] Carvalho, F., & Lima, A. (2020). Evaluation of machine learning techniques for cryptographic algorithm classification. Journal of Cryptology, 34(3), 507–523.
[7] Alharthi, A. M., & Alqahtani, A. S. (2020). A comparative study of cryptographic algorithm identification using statistical features. International Journal of Security and Its Applications, 14(2), 35–45.
[8] Zhu, Z., & Yang, J. (2019). Identification of encryption algorithms based on machine learning classifiers: A survey. Journal of Information Security, 10(2), 96–107.
[9] Srinivasan, R., & Deepika, K. (2017). Cryptographic algorithm classification using machine learning methods. International Journal of Computer Science and Network Security, 17(4), 125–130.
[10] Kumar, N., & Kaur, P. (2018). A machine learning approach for the classification of cryptographic algorithms. International Journal of Computer Applications, 181(4), 37–43.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details