CRYPTOGRAPHIC ALGORITHMS IDENTIFICATION BASED ON DEEP LEARNING

Ruiqi Xia[1], Manman Li[2] and Shaozhen Chen[2]

[1]Department of Cyberspace Security,
Information Engineering University, Zhengzhou, China
[2]State Key Laboratory of Mathematical Engineering and Advanced Computing,
Kexue Avenue, Zhengzhou, China

ABSTRACT

The identification of cryptographic algorithms is the premise of cryptanalysis which can help recover the keys effectively. This paper focuses on the construction of cryptographic identification classifiers based on residual neural network and feature engineering. We select 6 algorithms including block ciphers and public keys ciphers for experiments. The results show that the accuracy is generally over 90% for each algorithm. Our work has successfully combined deep learning with cryptanalysis, which is also very meaningful for the development of modern cryptography and pattern recognition.

KEYWORDS

Deep learning, Cryptography, Feature engineering, Residual neural network, Ciphers identification.

1. INTRODUCTION

1.1. Motivation

Cryptography is widely used in privacy protection and communication security with the development of computer techniques [1]. The main target of cryptanalysts is to recover the keys from the ciphertext. However, they can only acquire ciphertext through public channels most of the time. Cryptanalysts determine the scopes of encryption algorithms through monitors, reverse analysis or Side Channel Attack of ciphers. They will know which several possible ciphers are used at the moment only based on ciphertext. Knowing the encryption algorithms assists cryptanalysts in recovering the keys. Therefore, it is very important to identify the encryption algorithms in advance for cryptanalysis.

1.2. Related Work

Thanks to the development of artificial intelligence, we can use machine learning techniques to solve the problems of cryptanalysis. Some researchers have already studied the identification of cryptographic algorithms. In 1998, Ramzan proposed that neural networks could be used for identifying ciphers [2]. Subsequently, Dileep, et. al [3-5]successfully identified DES, Blowfish and some other algorithms by Support Vector Machine and Decision Trees. However, the results became unsatisfactory when the keys were changed. Recently, Mishra, et. al [6,7] applied PART, C4.5 to the ciphers identification and the accuracy reached over 80%.

David C. Wyld et al. (Eds): ICAIT, CBIoT, WiMo, CRYPIS, ICDIPV, CAIML, NLCA - 2022
pp. 211-219, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.121217

Although deep learning has become very popular in many subjects, we notice that few scholars consider identifying the algorithms by deep neural network. In 2021, Sandeep, et. al tried to apply convolutional neural network to the identification of several block ciphers. However, the work did not show a detailed scheme[8]. On the other hand, the cryptosystem often uses random keys for safety while the previous work was unsatisfactory when the keys were unfixed. This makes the work less valuable in application. Therefore, there are many challenges for us to investigate further.

1.3. Our Contribution

It is necessary to investigate how to improve the accuracy of identification in the conditions of random keys and give a detailed and executable scheme based on deep learning. Hence in this paper we construct a novel model of cryptographic algorithms identification based on feature engineering and residual neural network. We select 6 algorithms including block ciphers and public keys ciphers for experiments. The accuracy is generally over 90% for each algorithm in the conditions of random keys. Compared with the former work, not only do we successfully apply the neural network to ciphers identification, but also improve the results of experiments in the conditions of random keys. Such technique assists cryptanalysts in recovering the keys and obtaining the plaintext. Our work also provides a new direction for the development of pattern recognition.

1.4. Arrangement

The arrangement of the paper is shown as follows. The first section introduces the background of our work and the main contribution. The second section briefly describes the cryptographic algorithms. We illustrate the model of identification in the third section, which includes feature engineering, residual neural networks and so on. The fourth section is the experiments of identification based on our approach. The last section is the conclusion.

2. CRYPTOGRAPHIC ALGORITHMS

Modern Cryptography could be divided into symmetric cryptography and asymmetric cryptography. These cryptographic algorithms make remarkable contribution to information security and privacy. In this work we select 4 block ciphers and 2 public keys ciphers for experiments. All of them are commonly used in reality.

2.1. Block Ciphers

Block ciphers divide the plaintext into fixed-length blocks and then encrypt or decrypt the encoded block sequences using the same keys. These algorithms are widely used in the protection of hardware, digital signature and so on. Nowadays, lightweight block ciphers become one of the most useful applications in IoT devices [9].

AES(Advanced Encryption Standard)[10]. The construction of AES is SPN (Substitution Permutation Network) structure. The block length is 128 bits. The key length is 128/192/256 bits. The numbers of rounds are 10/12/14. Here we use AES-128.

KASUMI [11]. The construction of KASUMI is Feistel structure with 64 bits block length and 128 bits key length. The number of rounds is 8. KASUMI algorithm was designed for the basis of the 3GPP (3rd Generation Partnership Project).

3DES (Triple Data Encryption Standard)[12]. 3DES was developed to overcome the shortages of DES. The block length is 56 bits and the key length is 168 bits.

PRESENT [13]. PRESENT belongs to lightweight block ciphers. The construction is SPN structure. The block length is 64 bits and the key length is 80/128 bits. The number of rounds is 31.

2.2. Public Keys Ciphers

Public keys ciphers encrypt the plaintext with the public key and decrypt with the private key. Public key cryptography was designed by Whitfield Diffie and Martin Hellman in 1976[14]. We use RSA and ElGamal algorithms for the research.

RSA [12]. RSA algorithm is based on the decomposition of large numbers. The private keys are computationally difficult to require from the public keys. The key length is usually 1024 bits or 2048 bits.

ElGamal[15]. ElGamal algorithm is another public key cryptography. It is based on calculating the discrete logarithm over a finite field. This algorithm is broadly used in digital signature.

3. MODEL OF IDENTIFICATION OF CIPHERS

3.1. Design for the Model

Figure 1 shows the construction of our model. The model of identification mainly consists of three parts with two stages: obtaining the original datasets, feature engineering, and deep neural network classifier. The ciphertext is encrypted by each algorithms in the conditions of random keys. After obtaining the ciphertext the feature engineering extracts the feature indices of it. Each feature vector is attached with the corresponding labels. Finally the feature files are inputted into the model for training and testing. The whole process includes training phase and testing phase. We package the whole process to form an end-to-end framework for application.

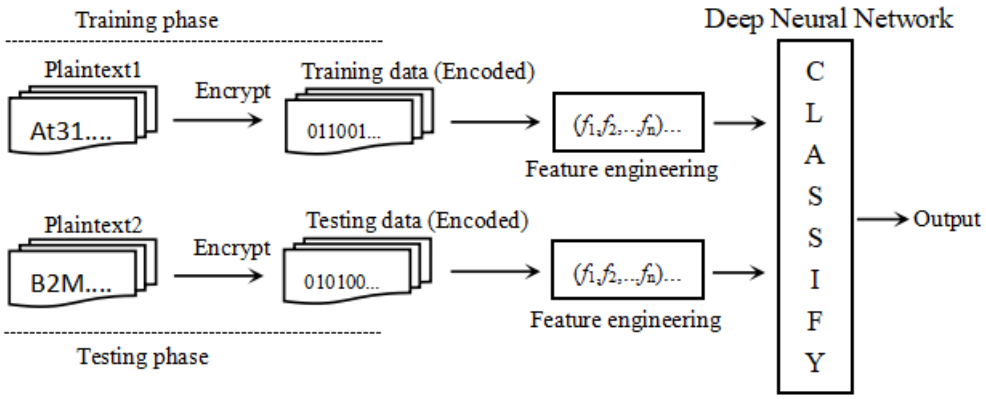

Figure 1. Model of Identification

In our experiments, 6 algorithms are encrypted by random keys. Meanwhile, the block ciphers are encrypted in CBC (Cipher Block Chaining) mode for safety. Compared with [6,7], the conditions of our experiments are far more strict, which make our work more meaningful.

3.2. Feature Engineering

Ciphertext seems diffused and random, especially when keys are unfixed. Feature engineering helps find out the characteristics of the data and helps the models work better. Although the deep neural networks could automatically extract the features of the data, we can make it in advance to help the networks understand the target and work more effectively. Feature engineering is one of the most essential steps in our work. We select three randomness indices published by NIST(National Institute of Standard and Technology) [16] in our experiments.

Frequency within blocks index. The frequency within blocks index collects the proportion of 0 or 1 in each sub-block divided from the ciphertext blocks.

Runs index. The index collects the sum of each length of run in the sequence. A run of length k consists of exactly k identical bits and is bounded before and after with a bit of opposite value.

Serial index. The index gets the sum of all sub-sequences of the ciphertext. A sequence of length m has 2^m sub-sequence. If there is no such sub-sequence, note it with 0.

Feature engineering extracts such three feature indices to make up the feature vectors. The corresponding labels are attached to each feature vectors. Then we input the feature vectors into the neural network for training and testing. The end-to-end framework we construct for application will package the feature engineering so it is more convenient to use the model.

3.3. Residual Neural Network

Thanks to the development of artificial intelligence, deep learning technology has been successfully combined with cryptography such as recovering the keys and simulation encryption [17]. In 2019, Gohr applied residual neural network to cryptanalysis [18], which improved the traditional cryptanalysis significantly. Inspired by his work, we also use such neural network for ciphers identification.

Residual neural network introduces a residual tower which helps the model works better when the depth of network increases. Such networks avoid degradation by using identity mappings [19]. Here we choose ”ReLU” as the activation and ”Conv1D” as the basic convolution layer. Most importantly, we use cross entropy as the loss function. It is shown as follow(y_i means the real value and a_i means the prediction value).

$$\text{cost} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \ln a_i + (1-y_i)\ln(1-a_i)\right] \tag{1}$$

Figure 2 shows the structure of residual neural network.

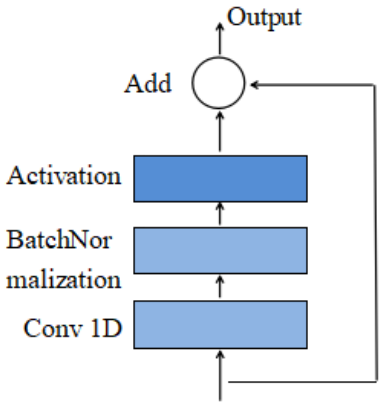

Figure 2. Residual neural network

4. EXPERIMENTS AND RESULTS

The hardware configurations of the experiments are Windows 10 system with 2 Intel Core i7 processors having 16 cores each and one NVidia GEFORCE RTX 2080Ti GPU, 256 RAM. The software we used is Python 3.7 with Keras 2.3.1.

The plaintext is selected from the texts in Open American National Corpus (OANC). We select some commonly used texts or sentences as the plaintext such as idioms. The size of plaintext is around 1.2 GB, which is slightly more than previous work. Then the plaintext files are divided into 1000 parts, which are about 1.1 MB. The keys are changed at each time of encryption.

After obtaining the ciphertext, we extract the feature indices of each part, which has about 2600 feature vectors. The feature vectors are attached with the corresponding labels. We use 0 to 5 to represent such 6 algorithms. Then the feature vectors are inputted into the neural network for training and testing. We set the epochs 200. The proportion of training sets and testing sets is 6:4. The learning rate is 0.01 and the batch size is 500.

4.1. Results

The results of the classifier are expressed by accuracy, precision and recall [20]. *TP* (True Positive) represents the number of right examples which are sentenced to right ones. *TN* (True Negative) represents the number of right examples which are sentenced to wrong ones. *FP* (False Positive) represents the number of wrong examples which are sentenced to right ones and *FN* (False Negative) represents the number of wrong examples which are sentenced to wrong ones. Hence accuracy means the ratio of all samples which are correctly sentenced in the entire dataset.

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{2}$$

Precision means the ratio of *TP* in the samples sentenced to be right.

$$precision = \frac{TP}{TP + FP} \tag{3}$$

Recall refers to the proportion of *TP* in the whole right samples.

$$recall = \frac{TP}{TP + FN} \tag{4}$$

The results are shown in Table 1 and Figure 3.

Table 1. Results of identification.

Algorithm(Label)	Precision	Recall	Accuracy
AES-128(0)	89.23%	72.1%	90.05%
KASUMI(1)	92.36%	66.38%	91.46%
3DES(2)	90.6%	69.05%	93.65%
PRESENT(3)	93.12%	65.28%	95.72%
RSA(4)	88.45%	74.51%	88.79%
ElGamal(5)	89.77%	74.21%	90.93%

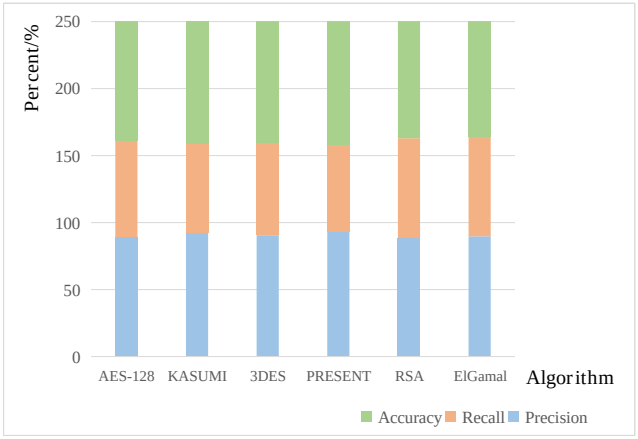

Figure 3. Percentage Stack histogram of identification

We find that the model works effectively in the experiments. Each algorithm's accuracy is higher than 90% generally. Even the worst group's accuracy is also higher than 85%. The precision of identification is also satisfactory, which is a little lower than the accuracy.

On the other hand, the recall is relatively lower, which is between 65% and 75%. According to the definitions of the three indices, when precision or accuracy becomes higher the recall becomes lower.

The average accuracy is 100/6 ≈ 17%. Therefore, our experiments' results are better than the average. This indicates that the model we construct is efficient indeed. Such technique can be used in practice.

The results of identification of block ciphers are better than public keys ciphers, which is around 5% higher in accuracy. The phenomenon means that block ciphers are more easily to be distinguished compared with public keys ciphers.

4.2. Discussion

The algorithms influence the results of identification significantly. Based on the results, we conclude that the block ciphers are more easily to be identified by classifiers. The ciphertext of

block ciphers may have more characteristics or differences which can be identified by neural networks. So we recommend that people should avoid using the algorithms which have many characteristics in the ciphertext for privacy safety.

Although we use random keys for encryption and CBC mode for block ciphers, the results of identification are still remarkable and stable. The gap between the highest accuracy and the lowest accuracy is less than 10%, which is more stable than previous work. It indicates that although deep neural network model needs more data, the model has stronger generalization ability which can be applied to more algorithms.

However, the recall seems unsatisfactory. Recall means the ability to find the correct samples in all correct sets, while precision means the ability to judge all correct samples. Hence our model ought to improve the ability in finding correct samples further. In addition, the model cannot give a correct classification when the ciphertext is not encrypted by the included algorithms. These shortages are worth improving in the future.

5. CONCLUSIONS

We study the identification of cryptographic algorithms in this work. The model of identification is based on the residual neural network and the feature engineering. The neural network classifier trains and tests the 3 feature indices extracted from ciphertext encrypted by 6 ciphers and random keys. Our experiments have successfully applied deep neural network to ciphers identification in detail. Compared with the previous work, not only do we improve the accuracy by around 10% in the conditions of random keys, but also investigate more complex ciphers including block ciphers and public keys ciphers.

Identifying cryptographic algorithms is one of the essential steps for keys recovery and it is useful in the application of cryptanalysis. According to Kerckhoffs' s assumption [21], cryptanalysts ought to know the encryption algorithms as well as other details. Therefore it is meaningful to identify the algorithms effectively in reality. Our work helps cryptanalysts know the encryption algorithms at the moment. So they will find out the most efficient method for recovering the keys more easily. It is also a novel application in pattern recognition, which provide some new directions for the development of deep learning.

In the future, we will possibly consider improving our model further. First, it is necessary to investigate whether we can reduce the size of data compared with the traditional machine learning approach. Second, the ability to judge all correct samples is still need to be improved. Meanwhile, it is necessary to apply our model to the identification of more cryptographic objects such as the modes of operation of block ciphers, etc. Hence there is much more research for identification of cryptographic algorithms.

ACKNOWLEDGEMENT

Thanks my fellows in State Key Laboratory of Mathematics and Advanced Computing! This paper is supported by Open Fund Project of the State Key Laboratory of Mathematical Engineering and Advanced Computing (No. 2019A08).

REFERENCES

[1] Coron, J. S. (2006). What is cryptography?. *IEEE security & privacy*, 4(1), 70-73.
[2] Ramzan, Z. (1998). On using neural networks to break cryptosystems. Manuscript.

[3] Dileep, A. D., & Sekhar, C. C. (2006, July). Identification of block ciphers using support vector machines machines. In The *2006 IEEE International Joint Conference on Neural Network Proceedings* (pp. 2696-2701). IEEE.

[4] Manjula, R., & Anitha, R. (2011, January). Identification of encryption algorithm using decision tree. In *International Conference on Computer Science and Information Technology* (pp. 237-246). Springer, Berlin, Heidelberg.

[5] Chou, J. W., Lin, S. D., & Cheng, C. M. (2012, October). On the effectiveness of using state-of-the-art machine learning techniques to launch cryptographic distinguishing attacks. In *Proceedings of the 5th ACM Workshop on Security and Artificial Intelligence* (pp. 105-110).

[6] Barbosa, F., Vidal, A., & Mello, F. (2016). Machine learning for cryptographic algorithm identification. *Journal of Information Security and Cryptography (Enigma)*, 3(1), 3-8.

[7] De Mello, F. L., & Xexeo, J. A. M. (2016). Cryptographic algorithm identification using machine learning and massive processing. *IEEE Latin America Transactions*, 14(11), 4585-4590.

[8] Pamidiparthi, S., & Velampalli, S. (2021). Cryptographic algorithm identification using deep learning techniques. In Evolution in Computational Intelligence (pp. 785-793). Springer, Singapore.

[9] Hatzivasilis, G., Fysarakis, K., Papaefstathiou, I., & Manifavas, C. (2018). A review of lightweight block ciphers. *Journal of cryptographic Engineering*, 8(2), 141-184.

[10] Abdullah, A. M. (2017). Advanced encryption standard (AES) algorithm to encrypt and decrypt data. *Cryptography and Network Security*, 16, 1-11.

[11] Kim, H. W., Park, Y. J., Kim, M. S., & Ryu, H. S. (2002). Hardware implementation of the 3GPP KASUMI crypto algorithm. In *Proceedings of the IEEK Conference* (pp. 317-320). The Institute of Electronics and Information Engineers.

[12] Singh, G. (2013). A study of encryption algorithms (RSA, DES, 3DES and AES) for information security. *International Journal of Computer Applications*, 67(19).

[13] Bogdanov, A., Knudsen, L. R., et. al. (2007, September). PRESENT: An ultra-lightweight block cipher. In *International workshop on cryptographic hardware and embedded systems* (pp. 450-466). Springer, Berlin, Heidelberg.

[14] Diffie, W., & Hellman, M. E. (2019). New directions in cryptography. In *Secure communications and asymmetric cryptosystems* (pp. 143-180). Routledge.

[15] ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4), 469-472.

[16] Soto, J. (1999, October). Statistical testing of random number generators. In *Proceedings of the 22nd national information systems security conference* (Vol. 10, No. 99, p. 12). Gaithersburg, MD: NIST.

[17] Benamira, A., Gerault, D., Peyrin, T., & Tan, Q. Q. (2021, October). A deeper look at machine learning-based cryptanalysis. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 805-835). Springer, Cham.

[18] Gohr, A. (2019, August). Improving attacks on round-reduced speck32/64 using deep learning. In *Annual International Cryptology Conference* (pp. 150-179). Springer, Cham.

[19] Thorpe, M., & van Gennip, Y. (2018). Deep limits of residual neural networks. *arXiv preprint* arXiv:1810.11741.

[20] Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint* arXiv:2010.16061.

[21] Xuejia, L. A. I. (2001). Basic Concepts of Cryptography. *Advanced Security Technologies in Networking*, 178, 21.

AUTHORS

Ruiqi Xia. Graduate student at the Institute of Cyberspace Security, Information Engineering University.

Manman Li. PhD of State Key Laboratory of Mathematical Engineering and Advanced Computing.

Shaozhen Chen. Professor of State Key Laboratory of Mathematical Engineering and Advanced Computing.