CRYPTO-MINE: Cryptanalysis via Mutual Information Neural Estimation

Benjamin D. Kim, Vipindev Adat Vasudevan, Jongchan Woo, Alejandro Cohen,
Rafael G. L. D'Oliveira, Thomas Stahlbuhk, and Muriel Médard

Abstract—The use of Mutual Information (MI) as a measure to evaluate the efficiency of cryptosystems has an extensive history. However, estimating MI between unknown random variables in a high-dimensional space is challenging. Recent advances in machine learning have enabled progress in estimating MI using neural networks. This work presents a novel application of MI estimation in the field of cryptography. We propose applying this methodology directly to estimate the MI between plaintext and ciphertext in a chosen plaintext attack. The leaked information, if any, from the encryption could potentially be exploited by adversaries to compromise the computational security of the cryptosystem. We evaluate the efficiency of our approach by empirically analyzing multiple encryption schemes and baseline approaches. Furthermore, we extend the analysis to novel network coding-based cryptosystems that provide individual secrecy and study the relationship between information leakage and input distribution.

Index Terms—Mutual Information, Cryptography, Individual Secrecy, Input Distribution, Machine Learning.

I. INTRODUCTION

Mutual information (MI) has a long history in cryptography, dating back to the era of Claude Shannon, who introduced the concept of information leakage and its relationship to secure communication systems [1]. According to the definition of perfect secrecy [1], the MI between the ciphertext and the plaintext is zero (i.e., *a posteriori* probability and *a priori* probability of finding the plaintext remains the same even if the ciphertext is known). However, achieving perfect secrecy requires as large a key as the message which is difficult in practice. Various relaxations from this condition have been explored, including computational security against adversaries with limited resources [2], [3] and information-theoretic security, where Eve's access to information is restricted [4]–[6]. Further deviations, such as individual secrecy, have also been widely explored [7]–[9]. Recent research combining information-theoretic methods and computational security notions without compromising communication rate has been shown to provide energy-efficient secure communication systems by linear coding across the data of multiple links and encrypting a portion of it [10]–[12].

However, these relaxations on the perfect secrecy condition result in information leakage between the ciphertext and plaintext. The leaked information from a cryptosystem can be used to evaluate its strength, particularly against side-channel attacks that exploit the physical properties of the implementation, such as power consumption and time. Although extensive research has been conducted on MI analysis on side channels to evaluate the security of cryptographic systems [13], [14], application of MI between the plaintext and ciphertext directly in the context of chosen plaintext attack [15], [16] has not been explored as much. Inferring the MI between the plaintext and its ciphertext could potentially reduce the complexity of finding the key, and thus the security level of the cryptosystem [13].

Many security schemes, especially those providing individual secrecy, also rely on randomness and expect the input messages to be both independent and uniformly distributed. This ensures that patterns in plaintext do not aid eavesdroppers by helping them deduce information from the ciphertext. Encryption schemes like AES Electronic Cipher Block (ECB) mode [17] and those providing individual secrecy can be susceptible to vulnerabilities arising from such patterns, potentially leaking additional information that assists adversaries in learning more about the combination of input messages. Furthermore, using the same encryption key for multiple encryption instances, especially with large files, can lead to an increase in the mutual information between the plaintext and ciphertext. From an information theory perspective, the only missing element is the key itself. Nevertheless, when dealing with properly uniform data and employing secure encryption methods, the ciphertext may appear entirely uncorrelated with the plaintext. Thus, the MI analysis between plaintext and ciphertext is of special interest in such cases but has been proven challenging due to their high-dimensional and nonlinear relationship. Recent advances in machine learning offer practical ways to estimate MI, using stochastic gradient descent over neural networks [18]. This estimation is useful for evaluating the strength of the cryptosystem since it can be exploited by malicious users.

In this article, we explore the use of MI estimation using neural networks to identify weak cryptosystems and their potential use in chosen plaintext attacks. Furthermore, we discuss the different assumptions from the information-theoretic

aspect of the cryptosystem and analyze the impact of input uniformity on security. We also use the novel MI estimator using neural networks (MINE) [18] to check information leakage due to the partial encryption scheme presented in the Hybrid Universal Network Coding Cryptosystem (HUNCC) [10], [19] and present the necessary guidelines for its practical use. We test the efficiency of such estimators in identifying the correlation between the plaintext and ciphertext without access to the key and its application as a tool for evaluating the security of cryptographic systems.

II. NEURAL ESTIMATION OF MUTUAL INFORMATION ON CRYPTOSYSTEMS

MI between variables with unknown distributions has been prohibitively difficult to calculate, making it an even bigger challenge to calculate MI for a finite dataset of high dimensional plaintext ciphertext pairs. However, the neural estimation of MI proves to be converging to a lower bound as the number of samples goes to infinity [18, Theorem 2]. This process follows the notion that MI between two random variables X and Y can be expressed by the Kullback-Leibler (KL) Divergence, the distance between their joint distribution and the product of their marginal distributions.

$$I(X;Y) = D_{KL}(P(X,Y)||P(X)P(Y))$$

Donsker-Varadhan representation of KL divergence is used to estimate MI, where Ω is the product sample space of the distributions P_1 and P_2, and the supremum is taken over all functions F, with a finite expectation. As seen in [18], F can be modeled as a neural network F_ϕ, where ϕ is optimized such that a maximum of $I_\phi(X;Y)$ can be computed using stochastic gradient descent (SGD) [20].

$$D_{KL}(P_1||P_2) = \sup_{F:\Omega\to\mathbb{R}} E_{P_1}[F] - \log(E_{P_2}[e^F])$$

$$I_\phi(X;Y) = E_{P(X,Y)}[F_\phi] - \log(E_{P(X)P(Y)}[e^{F_\phi}])$$

However, challenges have arisen from high variance with large MI estimations when solving the equation above. To mitigate this problem, [21] added a stabilization term to fix this problem.

$$I_\phi(X;Y) = E_{P(X,Y)}[F_\phi] - \log(E_{P(X)P(Y)}[e^{F_\phi}]) \\ - 0.1(\log(E_{P(X)P(Y)}[e^{F_\phi}]))^2$$

This regularization term helps the optimizer find one solution for our estimation, rather than wandering between a class of several functions. A more detailed discussion on the MI estimation can be found in [18], [21].

A. Neural Estimation Procedure

Mutual information neural estimation has been successfully used in different applications and optimizations of MI [18], [22], [23] and we demonstrate the estimator's capabilities with cryptographic protocols. To set the baseline, a few extreme cases are considered in section II-B and the experiments are further extended to several popular cryptographic protocols in section II-C.

Algorithm 1 MI Estimation for Cryptosystems
1: **Input** Plainext **X** for **N** samples
2: **Encrypt(x)** for ciphertext **Y** for **N** samples
2: Initialize network parameters θ
4: **repeat**
5:     Find **I(X;Y)** between the sample set
6:     Compute SGD optimizing and updating θ
7: **until** convergence

The estimation of MI in our experiments is achieved using the gradient descent of a neural network that takes the batches of the plaintext-ciphertext pairs as the input [18], [21], as shown in alg. 1. Specifically, the network consists of two intermediate layers of 100 nodes and an output node providing the MI. The number of input nodes depends on the dimensions of the input to the neural network. Our baseline experiments require 32 nodes in the input layer. We use ReLU non-linearity between layers. For each estimation, we use 100,000 samples. The plaintext and ciphertext each have a length of 16 bytes in our initial set of experiments. We use batch size 10,000, learning rate 1e-4, and 2000 or 5000 epochs, depending on the complexity of the protocol.

B. Baseline Experiments

For our baseline experiments to analyze the efficiency of the neural estimation, we deploy it on a few baseline scenarios such as plaintext and ciphertext being exactly the same (No encryption), one-time pad encryption, one-time pad with key, and simple XOR with the same key. We can get the true MI of no encryption setting, by using $I(X;Y) = H(X) - H(X|Y)$, where H is the Shannon entropy. Consider X to be our plaintext and Y to be our ciphertext. Since we do not encrypt, $X = Y$, and by definition $H(X|Y) = 0$ when $Y = X$. Since our inputs are uniformly generated, we can calculate $I(X;Y) = H(X) \simeq 11.09$ nats[1] and use this as an upper bound and performance indicator. Similarly, for a completely uncorrelated plaintext and ciphertext scenario, $H(X|Y) = H(X)$ and $I(X;Y) = 0$ by principle. However, the estimator is able to optimize the MI over the finite dataset and estimates $I_\phi(X;Y)$ a value negligibly greater than zero.

Our experiments with neural estimation for no encryption setting provide the MI estimation of $I_{NO}(X;Y) = 9.7$ nats converging around 100 epochs, as shown in Fig. 1. This shows that the MI estimation of our 16 dimension variables performs with great accuracy. The MI estimation of the repeating key XOR results in $I_{XOR}(X;Y) = 7.8$ nats. This high MI estimation matches our expectations since XOR-ing with the same key is a simple operation that results in a high correlation between the two variables. For our baseline estimation of a strong encryption scheme, we use a one-time pad (OTP), proven to provide perfect secrecy, i.e., zero MI between plaintext and ciphertext [1]. The MI estimation for the OTP is approximately 0.05 nats. However, if we add the key used

---

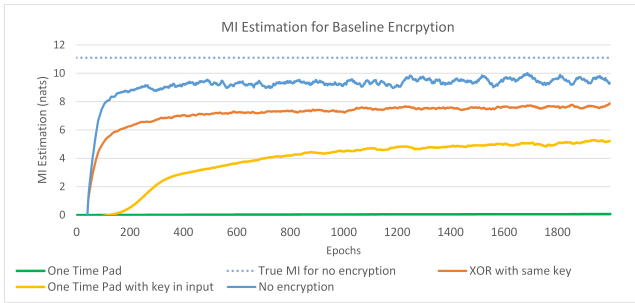[1]Our estimator uses natural logarithm

Fig. 1. Baseline MI estimation results

Fig. 2. MI estimation for popular cryptographic protocols

in OTP to the input, the estimator is easily able to detect the correlation of X and Y, as shown in Fig.1.

C. MI estimation for popular cryptographic protocols

We now use our estimator on several some well-known encryption schemes, including AES ECB, AES Counter Mode (CTR), a single SPN block cipher, and Caesar cipher (stream cipher). We again use the same parameters for the neural network, but due to the complexity of the operations in these encryption schemes, we run the estimator for 5000 epochs.

Fig. 2 depicts the results. As can be seen, the block cipher encryption scheme converges to an MI estimate of $I(X;Y) = 1.1$ nats, while the stream cipher encryption scheme converges to an estimate of around $I(X;Y) = 3.1$ nats. However, estimation over AES ECB and CTR modes result in a very low value, $I(X;Y) = 0.07$ nats. Interestingly, if we try AES ECB with non-uniform, correlated inputs, we estimate an MI of $I(X;Y) = 2.1$ nats. MI leakage from AES ECB implies that the estimator is able to rightly identify leakage of correlation between the plaintext and ciphertext, even though for any particular instance the encryption is secure.

III. MUTUAL INFORMATION ANALYSIS OF NETWORK CODING BASED CRYPTOSYSTEMS

Cryptosystems that guarantee individual secrecy through coding schemes combined with information theoretic approaches have been of particular interest in achieving secure communication with high data rates. Such systems may leak information about combinations of inputs while protecting each individual message from being decrypted. In this section, we analyze the security of a recently proposed Hybrid Universal Network Coding Cryptosystem (HUNCC) that provides *individual computational security* through coding and partial encryption [10]. This approach is best explained in a network setting with n messages to be sent over n communication links. The messages are linearly encoded using a generator matrix $\mathbf{G} \in F_{2^n}^{n \times n}$ using a network coding scheme [8], [9] before encrypting a subset of the outgoing links with any particular cryptosystem. By encrypting a small portion of the outgoing message, this approach achieves individual and computational security as presented in [10, Theorem 1] and individual indistinguishability under chosen ciphertext attack (IND-CCA1) as in [19].
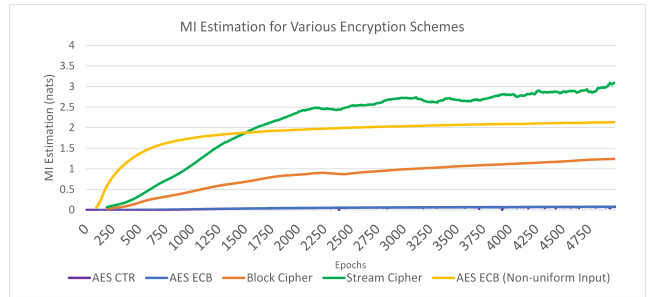
The MI analysis of this scheme is particularly interesting since it uses well-known cryptosystems to achieve computational security of individual messages with the random linear encoding of inputs. Our neural network estimator models a computationally limited adversary that estimates the MI. Furthermore, there are a few similar works that follow the same principle of linear encoding, but instead of cryptography using physical layer security, to achieve absolute physical layer security in high-frequency communication systems [24]. The results from this analysis can also be extended to such works with minor modifications.

In this section, we empirically examine these claims and examine HUNCC under different input distributions to analyze the necessary uniformity in HUNCC's input distribution. For this section, the encoding operation for HUNCC follows random linear coding [8] and the encryption scheme is AES-128 ECB, though HUNCC's universality allows almost any encryption scheme after encoding.

A. Analysis on HUNCC for Individual Secrecy

For the initial analysis of the HUNCC scheme, we consider 100,000 samples of uniformly distributed plaintexts of 128 bits, defined in $GF(2^8)$ as 16 bytes. There are 8 outgoing links with only one link encrypted using AES-128 ECB mode. The MI estimate of this setting is indistinguishable from a completely AES-ECB encrypted case, as shown in Fig. 4, for a uniformly distributed input. We also analyze the individual indistinguishability of the HUNCC scheme as defined in [19]. For this analysis, one particular input message among the 8 input plaintexts is set to a non-uniform input (as 128 bits of 1 in our case) and the rest of the inputs are chosen as random. It was observed that the ciphertexts of each instance were entirely different, and the MI estimator was not able to distinguish any particular pattern, estimating an MI of $I(X;Y) = 0.052$ nats, similar to the case when inputs are uniformly distributed in fig. 4. This analysis supports the claims in [10], [19] about individual computational security and individual IND-CCA1 attacks.

B. HUNCC's Robustness to Input Distribution

We analyze the security of HUNCC under inputs lacking uniformity and compare it with its underlying cryptosystem. We verify this by testing our MI estimator on HUNCC under different levels of input uniformity, ranging from completely
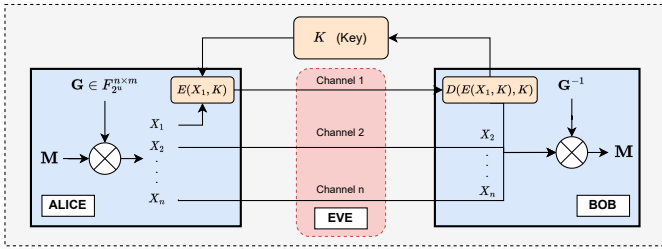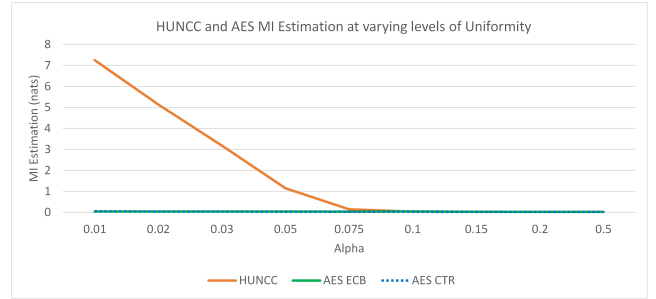
Fig. 3. Diagram of HUNCC

Fig. 4. Comparison between HUNCC and AES MI estimation for different levels of uniformity

TABLE I
MI ESTIMATION FOR DIFFERENT LEVELS OF UNIFORMITY (NATS).

α	HUNCC	AES CTR	AES ECB
0.01	7.2542	0.0384	0.0536
0.02	5.1399	0.0411	0.0411
0.03	3.1801	0.0395	0.0384
0.05	1.1455	0.0374	0.0263
0.075	0.1447	0.0354	0.0281
0.10	0.0319	0.0452	0.0298
0.15	0.0302	0.0258	0.0371
0.20	0.0233	0.0377	0.0334
0.5	0.0270	0.0359	0.0324

non-uniform all the way to completely uniform. We also test AES ECB and AES CTR full encryption for the same inputs to provide a comparison to the state-of-the-art encryption schemes.

To create inputs with such varying uniformity, we use a Gilbert-Elliot (GE) model [25], [26], with a 1 and 0 state, each generating their respective bit. This two-state Markov chain model with a state change probability of alpha (α) in both directions for simplicity. We vary α from 0.01 (non-uniform distribution) to 0.5 (uniform distribution) to introduce randomness in the input. For the experiments in this section, consider eight channels, each with 128 bits. Once these inputs are generated by our GE model, they are concatenated into 16 bytes per channel, just as the values were for the encryption schemes in section II. This leaves 128 symbols for the input (16 bytes over 8 channels). These inputs are then passed through our **G** encoding matrix, and then the first channel of the eight is encrypted by AES. For comparison, the uncoded inputs are fully encrypted with both AES ECB and AES CTR modes across all eight channels. For each estimation, we use the same parameters and architecture for our neural network as we did throughout section II, besides including 500,000 samples and 256 input nodes for the larger inputs.

Fig. 4 and table I illustrate the results of the complete set of tests. It is evident that the HUNCC system exhibits significant information leakage when the inputs are non-uniform. However, the MI between inputs and the HUNCC output decreases rapidly as the randomness of the input increases. In fact, HUNCC performance matches that of AES with only a small amount of randomness in the input, an alpha value of 0.1. This analysis suggests that the strict theoretical requirement of a uniformly distributed input may be relaxed when going up against attacks based on learning MI between the plaintext and ciphertext. From a practical point of view, this relaxed requirement can be satisfied by a lossless compression scheme, such as the Lempel–Ziv–Welch (LZW) compression [27]. Analyzing the entropy in the inputs to determine randomness, an LZW compressed input with α = 0.02 provides an average Shannon entropy of 1.33 nats while an α = 0.1 input (where MI estimation of HUNCC approximately matches AES), measures an average entropy of 1.52 nats. Since compression schemes are used widely in communication systems for reduced bandwidth usage, this can be achieved without incurring any additional cost.

IV. CONCLUSIONS

An accurate MI estimator capable of measuring the leakage between high-dimensional random variables, such as plaintext and ciphertext, can be advantageous in gauging the efficiency of a cryptosystem. Leaked information varies significantly depending on the cryptosystem and the input distribution. Our empirical analysis with MINE showcases the capability of the neural network-based estimator to identify leakage in weaker cryptosystems and its limitations in learning about stronger systems such as AES. This could be an important tool in the cryptanalysis of a security protocol to model a computationally limited adversary, e.g., in chosen plaintext attacks. Furthermore, we investigate how the input distribution impacts the security of cryptosystems, particularly those providing individual secrecy through a combination of linear coding and encryption, as in HUNCC. It is evident that highly correlated inputs result in leakage of information in such systems, but with proper uniformization of the input, the network coding-based cryptosystems limit their MI leakage to the same level as their underlying cryptosystems. Furthermore, our analysis shows that, for a practical application, lossless compression of the input will be sufficient to provide adequately uniform inputs.

REFERENCES

[1] C. E. Shannon, "Communication theory of secrecy systems," *The Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.
[2] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[3] J. Daemen and V. Rijmen, "AES proposal: Rijndael," 1999.

[4] A. D. Wyner, "The wire-tap channel," *Bell system technical journal*, vol. 54, no. 8, pp. 1355–1387, 1975.

[5] Y. Liang, H. V. Poor, and S. Shamai, *Information theoretic security*. Now Publishers Inc, 2009.

[6] M. Bloch and J. Barros, *Physical-layer security*. Cambridge University Press, 2011.

[7] K. Bhattad and K. R. Narayanan, "Weakly secure network coding," *NetCod, Apr*, vol. 104, 2005.

[8] D. Silva and F. R. Kschischang, "Universal weakly secure network coding," in *2009 IEEE Inf. Theory Works. on Net. and Inf. Theory*. IEEE, 2009, pp. 281–285.

[9] A. Cohen, A. Cohen, M. Medard, and O. Gurewitz, "Secure multi-source multicast," *IEEE Transactions on Communications*, vol. 67, no. 1, pp. 708–723, 2018.

[10] A. Cohen, R. G. L. D'Oliveira, S. Salamatian, and M. Médard, "Network coding-based post-quantum cryptography," *IEEE journal on selected areas in information theory*, vol. 2, no. 1, pp. 49–64, 2021.

[11] R. G. L. D'Oliveira, A. Cohen, J. Robinson, T. Stahlbuhk, and M. Médard, "Post-quantum security for ultra-reliable low-latency heterogeneous networks," in *MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM)*. IEEE, 2021, pp. 933–938.

[12] J. Woo, V. A. Vasudevan, B. Kim, A. Cohen, R. G. L. D'Oliveira, T. Stahlbuhk, and M. Médard, "CERMET: Coding for Energy Reduction with Multiple Encryption Techniques– It's easy being green," *arXiv preprint arXiv:2308.05063*, 2023.

[13] B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel, "Mutual information analysis: A generic side-channel distinguisher," in *Int. Works. on Crypto. Hardware and Embedded Systems*. Springer, 2008, pp. 426–442.

[14] E. Prouff and M. Rivain, "Theoretical and practical aspects of mutual information based side channel analysis," in *Applied Cryptography and Network Security: 7th International Conference, ACNS 2009, June 2-5, 2009. Proceedings 7*. Springer, 2009, pp. 499–518.

[15] W. Diffie and M. E. Hellman, "Special feature exhaustive cryptanalysis of the nbs data encryption standard," *Computer*, vol. 10, no. 6, pp. 74–84, 1977.

[16] R. C. Merkle and M. E. Hellman, "On the security of multiple encryption," *Communications of the ACM*, vol. 24, no. 7, pp. 465–467, 1981.

[17] M. Dworkin *et al.*, "Recommendation for block cipher modes of operation: methods for format-preserving encryption," *NIST Special Pub.*, vol. 800, p. 38G, 2016.

[18] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm, "Mutual information neural estimation," in *International conference on machine learning*. PMLR, 2018, pp. 531–540.

[19] A. Cohen, R. G. L. D'Oliveira, K. R. Duffy, and M. Médard, "Partial encryption after encoding for security and reliability in data systems," in *2022 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2022, pp. 1779–1784.

[20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[21] K. Choi and S. Lee, "Regularized mutual information neural estimation," 2020.

[22] H. Esfahanizadeh, W. Wu, M. Ghobadi, R. Barzilay, and M. Médard, "Infoshape: Task-based neural data shaping via mutual information," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[23] A. A. Atashin, B. Razeghi, D. Gündüz, and S. Voloshynovskiy, "Variational leakage: The role of information complexity in privacy leakage," in *Proceedings of the 3rd ACM Workshop on Wireless Security and Machine Learning*, 2021, pp. 91–96.

[24] A. Cohen, R. G. L. D'Oliveira, C.-Y. Yeh, H. Guerboukha, R. Shrestha, Z. Fang, E. Knightly, M. Médard, and D. M. Mittleman, "Absolute security in terahertz wireless links," *IEEE Journal of Selected Topics in Signal Processing*, 2023.

[25] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell sys. tech. journ.*, vol. 39, no. 5, pp. 1253–1265, 1960.

[26] E. O. Elliott, "Estimates of error rates for codes on burst-noise channels," *The Bell Sys. Tech. Journ.*, vol. 42, no. 5, pp. 1977–1997, 1963.

[27] T. A. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, no. 06, pp. 8–19, 1984.