Postprint: Fuzzy-Key Cryptographic Communication Based on Generative Adversarial Networks

Authors: Ximing Li, Wu Jiarun, Wu Shaoqian, Guo Yubin, Marsha

Date: 2019-05-10T00:00:00+00:00

Abstract

Fuzzy key encryption communication refers to achieving secure encrypted communication when the keys held by communicating parties possess certain discrepancies. Generative Adversarial Network (GAN) is a novel framework for obtaining generative models through adversarial learning; through the adversarial interplay between generative and discriminative models, it enables accurate estimation of sample data distributions, and has been utilized to achieve secure communication in the presence of adversaries. The objective is to address the problem of fuzzy key encryption communication, and to preliminarily implement a fuzzy key encryption communication scheme under symmetric keys using the Generative Adversarial Network methodology. Initially, neural networks are employed to implement two-party fuzzy key encryption communication, achieving fuzzy key encryption communication with 6-bit key discrepancy within 16-bit symmetric key encryption communication. Building upon this, a fuzzy key encryption communication model considering the presence of adversaries is constructed, utilizing GAN principles to conduct adversarial training between the communicating parties and the adversary, thereby achieving fuzzy key communication with 4-bit key discrepancy within 16-bit symmetric key encryption communication. In the experimental model, the communicating parties can communicate normally, while the adversary is unable to obtain plaintext information even when capable of acquiring ciphertext. The experiments demonstrate the feasibility of employing neural networks and Generative Adversarial Networks to solve the fuzzy key encryption communication problem.

Full Text

Preamble

Vol. 37 No. 6
Application Research of Computers
ChinaXiv Partner Journal

Study on Fuzzy Key Encryption Communication Based on Generative Adversarial Networks

Li Ximing, Wu Jiarun, Wu Shaoqian, Guo Yubin†, Ma Sha
(School of Mathematics & Informatics, South China Agricultural University, Guangzhou 510642, China)

Abstract: Fuzzy key encryption communication refers to secure encrypted communication between parties whose keys contain certain differences. Generative adversarial networks (GAN) represent a novel framework for obtaining generative models through adversarial learning, where the adversarial game between generative and discriminative models enables accurate estimation of sample data distributions. GANs have been utilized to achieve secure communication in the presence of adversaries. This paper addresses the problem of fuzzy key encrypted communication and preliminarily implements a fuzzy key encryption communication scheme under symmetric keys using GANs. First, we employ neural networks to realize two-party fuzzy key encryption communication, achieving secure communication with 6-bit key differences in 16-bit symmetric key encryption. Building upon this, we consider an adversarial model for fuzzy key encryption communication, utilizing GAN principles to conduct adversarial training between communicating parties and adversaries. This achieves fuzzy key communication with 4-bit key differences in 16-bit symmetric key encryption. In the resulting model, legitimate parties can communicate normally while adversaries cannot obtain plaintext information even when accessing ciphertext. Experiments demonstrate the feasibility of solving fuzzy key encryption communication problems using neural networks and generative adversarial networks.

Keywords: generative adversarial networks; fuzzy key encryption; batch normalization; fully-connected neural network; convolutional neural network

---

0 Introduction

Generative adversarial networks (GAN) are a type of generative model comprising a generator and a discriminator. The core idea involves adversarial learning between generative and discriminative models through a competitive game, where both components continuously strengthen each other during mutual opposition, ultimately yielding a generative model capable of producing highly realistic data. Since Goodfellow et al. [**?**] proposed GAN, its favorable

characteristics have led to rapid application across various domains with remarkable results. The most extensive applications appear in computer vision, including image generation and segmentation [**?**], image style transfer [**?**], and others. Additionally, notable achievements have been realized in information retrieval [**?**] and text generation [**?**].

In 2016, Google Brain' s Abadi et al. published research on protecting secure communication using adversarial neural networks [**?**]. This encrypted communication model consists of two communicating neural networks, Alice and Bob, and an eavesdropper neural network, Eve. When Alice and Bob conduct encrypted communication, they aim to limit the information Eve can obtain from intercepting their communications. During training, Alice and Bob strive to increase encryption and decryption complexity while ensuring accurate plaintext encryption and decryption, whereas Eve attempts to make its decryption results as close to the plaintext as possible, thereby improving decryption accuracy. Through adversarial training, an encrypted communication model can be obtained that ensures normal information exchange while resisting external eavesdropping.

Fuzzy key encryption communication refers to an encryption scheme that guarantees normal communication when certain differences exist between the keys of communicating parties, while simultaneously preventing adversarial eavesdropping and attacks. Such schemes appear in numerous encrypted communication application scenarios. For instance, in biometric-based encrypted communication, fingerprint or iris information collected during decryption differs from the originally captured data, requiring current encryption-decryption systems to employ extractor technologies to generate unique keys before correct decryption becomes possible [**?**][**?**]. Similarly, in fuzzy identity-based encryption, if the identity information used for encryption is "university faculty," decryption should also succeed using identities such as "university staff" or "professor," meaning the keys of both parties contain certain differences [**?**][**?**].

Implementing fuzzy key encryption communication using traditional algorithmic approaches proves extremely difficult. However, Abadi et al.' s perspective of treating secure communication parties as neural networks and employing GANs for adversarial training enhances the feasibility of designing and implementing fuzzy key encryption communication schemes. This paper first utilizes Abadi et al.' s neural network model for fuzzy key encryption communication. Experiments demonstrate that their proposed neural network cannot directly achieve fuzzy key encryption communication without considering adversaries, yet Bob, the decrypting party, obtains more information with key differences than Eve in Abadi et al.' s experiments, who had no key at all.

Based on these findings, this paper designs a two-party fuzzy key encryption communication model composed of Alice and Bob, modifying Abadi et al.' s neural network model. By adding fully connected layers, modifying activation functions, and applying data batch normalization, we achieve secure communication with 6-bit key differences under 16-bit symmetric key conditions. Subse-

quently, using GAN principles, we achieve secure communication with 4-bit key differences in the presence of adversaries.

1.1 Using GAN to Achieve Secure Encrypted Communication

In their paper "Learning to protect communications with adversarial neural cryptography" [**?**], Abadi et al. treat both communicating parties (Alice and Bob) and the adversary (Eve) in symmetric encryption models as neural networks, utilizing GAN to achieve secure encrypted communication in adversarial settings.

Their work begins with a classic cryptographic scenario, as shown in Figure 1 [Figure 1: see original paper]. Alice sends a message to Bob. Alice first encrypts the plaintext P using key K to obtain ciphertext C. Both Bob and Eve can receive C. Bob decrypts C using key K to recover the message, while Eve, lacking the key, can only attempt decryption through neural network training to obtain message . The encryption-decryption model composed of Alice and Bob and the adversarial model composed of Eve learn through continuous adversarial training, ultimately making while maximizing the difference between and P.

Abadi et al. employed identical neural network architectures for Alice, Bob, and Eve, with different inputs and outputs, as shown in Figure 2 [Figure 2: see original paper]. Alice uses this neural network for encryption, with inputs P and K (the leftmost input group in Figure 2) and output C. Bob performs decryption with inputs C and K (the rightmost input group in Figure 2) and output . Eve performs decryption with input C (the middle input group in Figure 2) and output . Through adversarial training, Alice' s encryption capability continuously strengthens, and decrypting party Bob correspondingly improves its decryption ability, achieving = P. Meanwhile, Eve' s decryption result differs from the original text by 7-8 bits in the 16-bit plaintext scenario. Since each bit takes values of -1 or 1, Eve' s decryption results resemble randomly generated bits, preventing acquisition of useful information through adversarial training.

1.2 Fuzzy Key Encryption Communication Testing

This paper employs Abadi et al.'s neural network (see Figure 2) to test fuzzy key encryption communication, assessing the feasibility of using neural networks for this purpose. The test first considers only fuzzy key encryption communication between Alice and Bob, temporarily removing Eve. Alice' s inputs and outputs remain unchanged. In Bob' s inputs, ciphertext C remains the same while key K becomes K' , representing a fuzzy key differing from the original key K, with output still denoted as .

Figure 2 illustrates the neural network structure for Alice, Bob, and Eve. Alice's input consists of plaintext P and key K, while Bob' s input comprises ciphertext C and fuzzy key K' , where P, K, and K' are all represented as arrays of length N (N = 16 in testing). Each bit in P and K randomly takes values of -1 or 1. We randomly select n bits from key K and invert them to generate fuzzy key K' , where n represents the number of key difference bits.

In experiments, both Alice and Bob use the AdamOptimizer for model optimization. Training proceeds for 10 rounds, with neural networks Alice and Bob each iterating M1 times per round (M1 = 2000), and the learning rate set to 0.0008 for both models.

To evaluate decryption loss caused by fuzzy keys, this paper references the loss calculation method from [**?**] and proposes the loss rate formula:

$$L = \frac{1}{2N} \sum_{i=1}^{N} |P_i - \hat{P}_i|$$

where N is the length of P. Since plaintext bits take values of 1 or -1, the loss rate measures the percentage of bits where differs from P.

We tested Bob' s decryption performance with key differences of 1 bit and 2 bits, with results shown in Figure 3 [Figure 3: see original paper].

Figure 3 shows that when key differences are 1 or 2 bits, experimental results are nearly identical, with Bob' s loss rate at approximately 0.25, meaning the gap between Bob' s decrypted plaintext and plaintext P is about 4 bits. Increasing the key difference from 1 to 2 bits has almost no impact on Bob' s decryption loss rate. This indicates that Bob cannot correctly decrypt when key differences exist, but the loss rate is significantly lower than Eve' s in Abadi et al.' s experiments (where Eve lost 7-8 bits, a loss rate of approximately 50%). This demonstrates that neural networks can use differential keys for feature recognition and influence experimental results. Therefore, improvements to Alice and Bob' s neural network models may enable fuzzy key communication.

---

2 Two-Party Fuzzy Key Encryption Model

Based on the above testing, this section improves the neural networks for Alice and Bob to implement two-party fuzzy key encryption communication. We present three aspects of neural network improvements along with experimental results and analysis.

2.1 Adding Fully Connected Neural Networks

To enhance analysis intensity for ciphertext and fuzzy keys, this paper first adds a fully connected layer to Bob' s decryption model, as shown in Figure 4 [Figure

4: see original paper]. For convenience, we designate this model as Bob1. Alice' s neural network model and encryption process remain unchanged, still using the model shown in Figure 2.

Using neural network model Bob1 for decryption training, we tested key differences of 1, 2, and 3 bits, calculating loss rates as before. Experimental results are shown in Figure 5 [Figure 5: see original paper].

The results demonstrate that Bob1 shows some improvement in fuzzy key discrimination capability. With key differences of 1 or 2 bits, Bob can completely decrypt ciphertext correctly. However, when key difference reaches 3 bits or more, model loss remains near 0.1, meaning differs from the 16-bit plaintext P by approximately 1 bit. Thus, Bob still cannot correctly decrypt plaintext with 3-bit key differences, but decryption loss is lower than in the Figure 2 test results.

2.2 Improving Activation Functions

Analysis of the model reveals that the Sigmoid activation function in Bob1's first fully connected layer is unfavorable for model weight updates. As noted in [**?**], the derivative of the Sigmoid function lies in the range (0, 0.25), reducing neural network information by 75% during backpropagation weight updates, with even greater impact after two Sigmoid functions. The tanh function' s derivative in the range (0, 1) enables more comprehensive utilization of information passing through the neural network. Therefore, we set the activation function of the first fully connected layer in model Bob1 to tanh, keeping the remaining Bob structure unchanged. For convenience, we designate the modified Bob1 model as Bob2.

Training neural network model Bob2 for decryption with key differences of 1-5 bits, we calculated loss rates as before. Experimental results are shown in Figure 6 [Figure 6: see original paper].

Figure 6 shows that with increasing training iterations, model Bob2 can correctly decrypt when key differences are below 4 bits, improving fuzzy key handling capability from 2 bits to 4 bits. However, when key difference reaches 5 bits, Bob' s loss rate approaches 0.25, meaning the gap between and P is about 4 bits.

2.3 Batch Normalization

Batch normalization [**?**] standardizes activation function distributions into linear regions, thereby increasing search step size, accelerating convergence speed, and reducing susceptibility to local optima. To improve neural network training convergence speed while preventing local optima, this paper adds batch normalization operations to both fully connected layers in Bob2, reducing the probability that different initializations prevent the model from finding minimum loss points and improving model stability.

We add batch normalization operations to both Bob1 and Bob2 while keeping model structures unchanged. During decryption training for Bob1 and Bob2, we tested key differences of 3, 4, 5 bits and 5, 6, 7 bits respectively, calculating loss rates. Results are shown in Figure 7 [Figure 7: see original paper] and Figure 8 [Figure 8: see original paper].

Figures 7 and 8 demonstrate that after adding batch normalization, both Bob1 and Bob2 improve their fuzzy key handling capability by 1 bit. Bob1 can decrypt fuzzy key ciphertexts with key differences up to 4 bits, while Bob2 can decrypt fuzzy key ciphertexts with key differences up to 6 bits.

---

3 Adversarial Fuzzy Key Encryption Model

This chapter employs generative adversarial network principles to train fuzzy key encryption communication models by incorporating the Eve model and treating communication parties Alice, Bob, and adversary Eve as opponents in training, aiming to obtain correct and secure encrypted communication schemes.

The neural network structure used in this section is shown in Figure 9 [Figure 9: see original paper]. For convenience, we designate this model as A-B-E. Alice continues using the neural network model shown in Figure 2, Bob uses model Bob2, and neural network Eve has the same structure as Bob2 but with ciphertext C as input and plaintext as output. All fully connected layers in Alice, Bob, and Eve structures include batch normalization operations.

Through training, Alice may generate ciphertexts that neither Eve nor Bob can understand. Continuous training of Alice and Bob for communication strengthens the two-party fuzzy key model, which is then adversarially trained with the Eve model. Through ongoing training, model A-B-E achieves normal communication between parties under high key differences while resisting adversary Eve.

During training, we again employ the AdamOptimizer for model optimization. Model A-B-E undergoes 60 training rounds, with neural networks Bob and Eve each iterating 2000 times per round and learning rates set to 0.005 for both. Experimental results are shown in Figure 10 [Figure 10: see original paper].

In the experiments, we randomly select n bits (n = 3, 4) from the key for fuzzification and use Equation (1) to calculate Eve's loss rate, i.e., . We calculate Alice and Bob' s loss rate through and :

$$L_{AB} = L_{Bob} + (1 - 2L_{Eve})$$

Equation (3) reflects that Alice and Bob aim to minimize Bob' s loss rate while maximizing Eve' s loss rate.

Figure 10 shows that after adding Eve, the model can effectively block Eve from cracking communication content while ensuring normal communication between Alice and Bob, though Bob' s capability to handle fuzzy key decryption decreases. With increasing training iterations, Alice and Bob can conduct secure communication and outperform Eve when key difference is 3 bits. However, when key difference reaches 4 bits, although the model can resist Eve' s attacks, Bob also cannot correctly decrypt ciphertext.

4 Conclusion

This paper addresses fuzzy key encryption communication problems using neural networks and generative adversarial networks. Our experiments are implemented under 16-bit symmetric key encryption communication environments. Results demonstrate that in two-party encrypted communication, neural network model improvements enable fuzzy key encryption communication with 6-bit key differences. In adversarial settings, fuzzy key encryption communication with 3-bit key differences can be achieved.

Our experiments remain preliminary and have not yet yielded fuzzy key encryption communication schemes based on generative adversarial networks for more complex and practical scenarios involving longer keys or asymmetric key encryption. However, this work demonstrates the feasibility of solving fuzzy key encryption communication problems using generative adversarial networks. Future work includes increasing key bit lengths, considering more complex and practical application scenarios, and developing practical fuzzy key encryption schemes.

References

[1] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks [EB/OL]. (2014-06-10) [2018-08-26]. https://arxiv.org/abs/1406.2661

[2] Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2017, 39(4): 640-651.

[3] Gatys L A, Ecker A S, Bethge M. Image style transfer using convolutional neural networks [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 2414-2423.

[4] Sunny S K, Angadi M. Potential roles and applications of thesauri in digital information retrieval systems [C]//Proc of the 5th International Symposium on Emerging Trends and Technologies in Libraries and Information Services. Piscataway, NJ: IEEE Press, 2018: 22-25.

[5] Hu Zhiting, Yang Zichao, Liang Xiaodan, et al. Toward controlled generation of text [C]//Proc of the 34th International Conference on Machine Learning. Sydney: PMLR Press, 2017: 1587-1596.

[6] Abadi M, Andersen D G. Learning to protect communications with adversarial neural cryptography [EB/OL]. (2016-10-24) [2018-08-26]. https://arxiv.org/abs/1610.06918.

[7] Li Ximing, Yang Bo. Construct a high performance fingerprint key extractor [J]. Computer Science, 2011, 38(3):107-11.

[8] Jiang Guanghan. Research on biometric encryption technology based on fingerprint [D]. Xi' an: Xidian University, 2017.

[9] Guo Fuchun, Susilo W, Mu Yi. Distance-based encryption: how to embed fuzziness in biometric-based encryption [J]. IEEE Trans on Information Forensics and Security, 2016, 11(2): 247-257.

[10] Ge Yangming. Research on user authentication and access control protocol for wireless sensor networks based on biometric keys [D]. Nanchang: Nanchang Hangkong University, 2018.

[11] Mehta G, Dutta M K, Kim P S. Biometric data encryption using 3D chaotic system [C]//Proc of the 2nd International Conference on Communication Control and Intelligent Systems. Piscataway, NJ: IEEE Press, 2016: 72-75.

[12] Li Ximing, Yang Bo, Guo Yubin, et al. A new key hiding scheme based on fingerprint [J]. Journal of Computer Research and Development, 2013, 50(3): 532-539.

[13] Wu Liqaing, Yang Xiaoyun, Han Yiliang. An efficient fuzzy identity encryption scheme based on ideal lattice [J]. Chinese Journal of Computers, 2015, 38(4): 775-782.

[14] Gianluca Fimiani. Supporting privacy in a cloud-based health information system by means of fuzzy conditional identity-based proxy re-encryption (FCI-PRE) [C]//Proc of the 32nd International Conference on Advanced Information Networking and Applications. Piscataway, NJ: IEEE Press, 2018: 569–572.

[15] Xu Bing, Huang Ruitong, et al. Revise saturated activation functions [EB/OL]. (2016-05-02) [2018-08-26]. https://arxiv.org/abs/1602.05980

[16] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift [EB/OL]. (2015-03-02) [2018-08-26]. https://arxiv.org/abs/1502.03167

Note: Figure translations are in progress. See original paper for figures.