International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 6 ǁ 2022 ǁ PP. 1788-1790

Cryptographic applications using artificial neural networks

Tanishq Vyas
department of computer science
Vellore institute of technology
Bhopal, Madhya Pradesh

Abstract*—Cryptography is keeping communications, information and important details safely using coding so that the person who is relevant to the details can access them. The techniques used to safeguard information in cryptography are derived from mathematical principles and a set of rule-based calculations known as algorithms that change signals in ways that make them difficult to decode. These algorithms are used to generate cryptographic keys, digitally sign documents, verify data privacy, browse the internet, and protect secret transactions such as credit card and debit card transactions. The goal of this research was to look into the usage of artificial neural networks (ANNs) in various types of digital circuits as well as in the field of cryptography. We investigated various neural network designs and training algorithms as part of our project. Different neural network topologies for an Adder are compared, and their benefits and drawbacks are highlighted.*

Keywords*—Artificial neural network, cryptography, mathematical principle, data privacy*

Date of Submission: 14-06-2022 Date of acceptance: 29-06-2022

I. Introduction-

Cryptography is concerned with the development of new security measures that protect anyone from trespasser reading. Data privacy systems are referred to as cypher systems. The cypher key is a set of rules created for the encryption of all news. Encryption is the process of converting open text, such as a message, into cypher text using rules. The inverse procedure, in which the receiver of the cypher turns it into the original text, is known as cryptanalysis of the news. Several important characteristics must be included in the cypher key. The singularity of encryption and cryptanalysis is the best example.

International alphabet letters, numerals, and punctuation marks are commonly used in the open text. The cypher text is composed in the same way as the open text.

II. PRELIMINARIES

A. Neural network

A neural network is an artificial intelligence strategy for teaching computers to analyse data in the same way that the human brain does. Deep learning is a sort of machine learning that employs interconnected nodes or neurons in a layered structure that closely resembles the human brain.

B. Dep learning

Deep learning is a subset of machine learning that is essentially a three-layer neural network. These neural networks aim to imitate the activity of the human brain by allowing it to "learn" from enormous amounts of data, albeit they fall far short of its capabilities.

III. NEURAL NETWORK AND CRYPTOGRAPHY

A Neural Network is a machine that is designed to model how the brain performs a task or function of interest. It can perform complex computations with ease. The objective of this project was to investigate the use of ANNs in various kinds of digital circuits as well as in the field of Cryptography. During our project, we studied different neural network architectures and training algorithms. A comparative study is done between different neural network architectures for an Adder and their merits/demerits are discussed. Using a Jordan (Recurrent network), trained by a back-propagation algorithm, a finite state sequential machine was successfully implemented. The sequential machine thus obtained was used for encryption with the starting key being the key for the decryption process. Cryptography was also achieved by a chaotic neural network having its weights given by a chaotic sequence.

C. Training the data

The neural network model is trained on encrypted data in a very basic method, without the overhead of interactive communication protocols or complicated security protocols), while simultaneously allowing privacy-preserving predictive analytics. The evolving functional encryption is the underlying crypto technique of our suggested CryptoNN. CryptoNN may perform a variety of authorized tasks on privacy-sensitive data encrypted by a distributed set of data.

This allows neural network models to be trained over encrypted data without the overhead of communication methods. The suggested CryptoNN framework is based on a secure matrix computation approach that uses current functional encryption for the inner-product scheme proposed by Abdalla et al. in our newly created functional encryption for basic arithmetic operations.

D. Deep learning for cryptography

Convolutional neural networks and recurrent neural networks are two deep learning (DL) architectures that show a lot of potential for artificial intelligence (AI) applications. Machine learning approaches based on neural networks have been successful in a variety of domains, including computer vision, speech/audio recognition, and others. To provide solid predictive analytics, such a neural network model requires a large amount of data to train. Commercial AI service providers like Google, Microsoft, and IBM have spent a lot of time and money developing deep learning models for a variety of intelligence applications, with the models being trained using data acquired from their clients. Even though AI-based applications make life easier, they also present severe privacy concerns due to the potential of extremely sensitive data being leaked.

Alice and Bob are unable to learn any strong encryption method since Eve only accepts C as an input, making it too weak to learn Alice and Bob's strong encryption. Finally, the scenario is the same in the third level, but there are two differences. First, Alice's input is from one of two plaintexts, denoted by SP = P1, P2, where P1 represents the first plaintext and P2 denotes the second plaintext; second, Eve's input is P1||P2||C, and Eve's outputs are the probability p0 that C is from P1 and the probability p1 that C is from P2.

Bob's network is similar to Alice's, except that it accepts C||K as an input. All of these changes are being made to obtain OTP. The design of Alice's network is shown in Fig. 4, where hi is neuron I in the output layer for a better understanding. Three stages are present in this piece. The finding reveals that Bob and Alice can learn to communicate where Bob can fully recover the plaintexts for every 4-bit, 8-bit, and 16-bit plaintexts in the first stage when there are only Alice's and Bob's networks.

This task is broken down into three stages. There are only Alice's and Bob's networks in the first stage, and the outcome reveals that Bob and Alice can learn to communicate, with Bob being to fully recover all 4-bit, 8-bit, and 16-bit plaintexts. However, the encryption algorithms they learn aren't secure, and they won't be able to accomplish OTP because the scenario doesn't include adversarial neural cryptography or Eve.

E. Secure Matrix Computation

The basic process in the training and prediction phases of neural networks is matrix computation. We build an encrypted matrix computation method, called secure matrix computation in the rest of the paper, by combining the functional encryption for the inner-product proposed in with our proposed functional encryption for the basic operations presented in Section III-B to support matrix computation over encrypted data.

Assume we need to compute a function f, where F is the allowed set of functions over the encrypted matrices X and Y that arrive from the client and server, respectively. Dot-product and elementwise arithmetic computation are among the permissible functions. Algorithm 1 explains the specific scheme. The scheme is divided into three sections: pre-process encryption, pre-process key derivative, and safe computation.

F. CryptoANN work

To demonstrate the scalability of CryptoNN, we employ another concrete and comparatively more sophisticated neural network model, namely a standard convolutional neural network for multiple classifications. The LeNet-5 model has five hidden layers: the convolutional layer (C1), the average pooling layer (S2), the convolutional layer (C3), the average pooling layer (S4), and the fully connected layer (S5), in addition to the input and output layers (C5).

We must concentrate on the C1 and output layers because our CryptoNN only focuses on the secure feed-forward and secure back-propagation / assessment processes in the model. In the secure feed-forward step, we must address the padding and convolution operations, and in the secure back-propagation/evaluation step, we must address the SoftMax output function and SoftMax cross-entropy loss function.

CryptoNN's security is examined. CryptoNN's underlying crypto scheme is the functional encryption system, whose security has already been discussed. We give a high-level security study of CryptoNN here. The authority is a third-party trustworthy authority that is not supposed to work with anyone. The server can only obtain the mapping relationship between the training data and the label from the server's perspective, while the

actual training/label data acquired from different data sources is safeguarded by functional encryption. Even for a basic label with high similarities, such as one label mapping to a set of training data, the encrypted output is evenly dispersed at random in the ciphertext space for each same label.

To be more exact, the prediction phase-only considers secure and normal feed-forward and obtains the neural network model's output. Because CryptoNN's trained model is plaintext, it can be used in conjunction with existing homomorphic encryption-based prediction systems. As a result, it offers a flexible range of privacy options, such as whether the projected label should be kept private or not. The HE-based prediction is used if the user chooses a confidential predicted label; otherwise, the FE-based prediction is used.

The x-axis indicates the element size (k) and the y-axis represents the processing time in MS in the matrix, where the x-axis represents the element size (k) and the y-axis represents the processing time in MS Pre-processing execution time for the function-derived key and element-wise computation has a linear characteristic as well. The computation of the secure element-wise and dot product is time-consuming, as indicated in the diagram. We use the parallelization technique in the implementation to address this issue. As illustrated in the parallelization technique, the secure computation may be made more practical by reducing the execution time of the secure dot-product from over 90 minutes to 8 seconds.

IV. CONCLUSION

Emerging neural networks based on machine learning methods such as deep learning and its variants have shown enormous promise in a variety of application domains that rely on or use large volumes of sensitive data from consumers. Several privacy-preserving machine learning algorithms have been presented in the literature to address the major privacy problems raised by the acquired data. These approaches leverage either secure multi-party computation or homomorphic encryption as the underlying mechanisms.

We present a novel CryptoNN framework for training neural networks over encrypted data using upcoming functional encryption techniques in this research. CryptoNN achieves the privacy goal as well as model correctness, according to the security analysis and performance evaluation.

REFERENCES

[1]. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, p. 436, 2015.
[2]. R. Bost, R. A. Popa, S. Tu, and S. Goldwasser, "Machine learning classification over encrypted data." in NDSS, 2015.
[3]. R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in International Conference on Machine Learning, 2016, pp. 201–210.
[4]. A. Mirhoseini, A.-R. Sadeghi, and F. Koushanfar, "Cryptoml: Secure outsourcing of big data machine learning applications," in Hardware Oriented Security and Trust (HOST), 2016 IEEE International Symposium on. IEEE, 2016, pp. 149–154.
[5]. B. D. Rouhani, M. S. Riazi, and F. Koushanfar, "Deepsecure: Scalable provably-secure deep learning," in 2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC). IEEE, 2018, pp. 1–6.
[6]. P. Mohassel and Y. Zhang, "Secureml: A system for scalable privacy-preserving machine learning," in 2017 38th IEEE Symposium on Security and Privacy (SP). IEEE, 2017, pp. 19–38.
[7]. R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in Proceedings of the 22nd ACM SIGSAC conference on computer and communications security. ACM, 2015, pp. 1310–1321.
[8]. M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016, pp. 308–318.
[9]. T. Graepel, K. Lauter, and M. Naehrig, "Ml confidential: Machine learning on encrypted data," in International Conference on Information Security and Cryptology. Springer, 2012, pp. 1–21.
[10]. F.-J. Gonzalez-Serrano, A. Amor-Mart ´ ´ın, and J. Casamayon-Ant ´ on, "Su- ´ supervised machine learning using encrypted training data," International Journal of Information Security, vol. 17, no. 4, pp. 365–377, 2018.
[11]. A. Weisberg and F. Armknecht, "Unsupervised machine learning on encrypted data," Cryptology ePrinArchiveve, vol. 2018, pp. Report–411, 2018.
[12]. E. Hesamifard, H. Takagi, and M. Ghasemi, "Cryptodl: Deep neural networks over encrypted data," arXiv preprint arXiv:1711.05189, 2017.
[13]. H. Chabanne, A. de Wargny, J. Milgram, C. Morel, and E. Prouff, "Privacy-preserving classification on deep neural network." IACR Cryptology ePrint Archive, vol. 2017, p. 35, 2017.