International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v12i1.7913
***Article Received**: 14 July 2023 **Revised**: 20 August 2023 **Accepted**: 01 September 2023*

CNN based Blockchain Information Protection Model for Emerging Cloud Applications

Chaitanya Kulkarni[1], Kamlesh Vasantrao Patil[2], Mugdha Arvind Rane[3], Ashwini Vitthal Kanade[4], Kavita Shantanu Sawant[5]

[1]Computer Engineering Department
VPKBIET, Baramati, India chaitanya.kulkarni@vpkbiet.org

[2]Department of Information Technology
Bharati Vidyapeeth's College of Engineering for Women Pune, India
kamlesh.patil@bharatividyapeeth.edu

[3]Department of Information Technology
Bharati Vidyapeeth's College of Engineering for Women Pune, India
mugdha.rane@bharatividyapeeth.edu

[4]Department of Information Technology
Bharati Vidyapeeth's College of Engineering for Women Pune, India
ashwini. kanade@bharatividyapeeth.edu

[5] Computer Engineering Department
Bharati Vidyapeeth's College of Engineering for Women Pune, India
kavita.sawant@bharatividyapeeth.edu

Abstract— In the age of mobile internet, the amount of data is growing, and ability to process service data is always getting better. So, protecting data privacy and making sure the service environment is trustworthy have become very important. This paper looks at a trusted privacy service computing model for common uses of convolutional neural networks. The goal is to find data and model calculation methods that support homomorphic encryption to protect data privacy. Build a service process certificate and a method for distributing calculation rights based on blockchain and new contract technology to make sure that service calculations are open, trustworthy, and easy to track. Explore how the new cloud environment resource data service model helps resource providers, model owners, and users work together to make the most of their resources and grow the sharing economy. Lastly, experiments are done to figure out how the model protects privacy.

Keywords- Blockchain, Data Privacy, Convolutional Neural Networks, Cloud Environment, Homomorphic Encryption .

I. INTRODUCTION

Human society has entered the era of mobile Internet, and intelligent computing, portable convenience, and private security have become important development trends. Therefore, strengthening the computing power of mobile terminals and improving the experience of high-intelligence computing services under the premise of protecting user privacy information is an urgent problem to be solved. Based Convolutional Neural Network (CNN) classifies images into scenes and explores critical technologies for new model applications. Currently, it is facing the following two challenges.

(1) User privacy protection is an essential prerequisite for applications. In 2018, the European Union formulated the General Data Protection Regulation [1] (General Data Protection Regulation, GDPR) proposes to strengthen the protection of personal data in terms of privacy and security. User terminal data involves much user information, and sending it directly to the cloud lacks security protection and risks leakage. Cloud service providers are also prone to excessive use of these data or private Sell, seek profit［2］. User data privacy protection is an essential requirement of the secure computing outsourcing model [3].

(2) Traditional cloud services are controlled and maintained by cloud providers, including service and rights rules, as well as transaction and service data. There is a lack of effective joint participation and management mechanisms, insufficient binding force and transparency, and it is challenging to pursue accountability when disputes arise. , it is easy for large cloud service providers to monopolize and difficult for small cloud service providers to survive, which is not conducive to the healthy development of the market and the effective integration of resources. To cope with the above challenges, this paper studies the CNN-oriented blockchain trusted privacy service, computing model. Taking typical applications as scenarios, using holomorphic encryption technology and effective use of

computing resources to provide computing power for edge devices under the premise of protecting user privacy Services, intelligent contracts, and blockchain IoMT is still more of a theoretical idea than a real solution, despite there being several alternatives. A faster adoption of cloud computing has an opportunity to create opportunities that will hasten the creation of clever solutions and reduce present constraints.

A. Homomorphic encryption technology

In 1978, Rivest first proposed the concept of holomorphic encryption; that is, the results of operations on ciphertext are equivalent to the effects of corresponding functions on the plaintext. The required calculation results can be obtained by processing ciphertext without decryption. This is the data. It is an essential means of privacy protection and has great significance [4~6].

In 2009, Gentry [7] proposed a fully holomorphic encryption scheme based on ideal lattices. However, due to the limitation of high complexity, the problem of cipher text data expansion cannot be effectively solved, which affects practical applications. Author [8] used the basic modular operation design Holomorphic encryption scheme (Dijk Gentry Halevi Vaiku-tanathan, DGHV), which is an improvement of the fully holomorphic encryption algorithm on integers in literature [7], reduces computational complexity, improves efficiency, and is easy to implement. One-time encryption of 1-bit data, the security of its public key encryption scheme depends on the "approximate greatest common divisor" problem.

In addition, literature [10] improved the scheme of literature [8] so that it can encrypt 2 bits of data at a time. Sun author [11] further improved the DGHV algorithm, expanding the plaintext space from 1 bit to n bits; a scheme that can encrypt n-bit data at a time is proposed, which reduces the number of encryptions. For clarity, this paper calls this algorithm N-DGHV. The N-DGHV algorithm has strong versatility and is suitable for service computing privacy protection scenarios, but there is still a public key: too much storage space. Therefore, based on the N-DGHV algorithm, this paper compresses and implements and optimizes the public key.

B. Maintaining Research on Cloud Computing Privacy

In the traditional cloud computing model, terminal data is transmitted to the cloud in plain text for calculation, user privacy cannot be guaranteed, and there are security risks [12]. Cloud computing data privacy protection solutions mainly include access control systems, data encryption, security outsourcing, secure multi-party computing, etc. [13] are based on data encryption theory. Author [14] proposed a secure multi-party computing method to solve the problem of cloud computing privacy protection. Participating in calculations and frequent communications is unsuitable for the application scenario where the client resources are limited. Literature [15] uses a privacy manager based on the obfuscation method to manage data in the cloud and user terminals to protect data privacy but focuses on encryption management in-depth study of intelligent calculation of cipher text and other work. Abbreviations and Acronyms

C. Other related research

Blockchain has the characteristics of distributed management and is difficult to tamper with; and can be widely used in many fields, such as medical care, transportation, agriculture, etc. Furthermore, smart contracts are usually open, and transparent calculation codes run on the blockchain. This paper uses the blockchain stores service data and designs an innovative contract equity evaluation model, which is unrestricted, transparent and automatically executed to enhance the credibility of the transaction. Federated learning technology was proposed in 2016. This technology is used to complete efficient and intelligent computing work to ensure data privacy and security. The process of federated learning is to calculate user data locally and transmit the results to the server to participate in aggregate calculations to protect the privacy data. However, the federated learning model data provider conducts model training locally, which requires high computing power in the terminal environment. Therefore, this paper selects the homomorphic encryption method based on the application scenario research work

II. TRUSTED PRIVACY SERVICE COMPUTING MODEL

To improve the service quality of convolutional neural network prediction in the cloud service environment, this paper studies the trusted privacy service computing model from three aspects: security, privacy and credibility. This model uses an asymmetric public key encryption and private key decryption mechanism to strengthen the security of data to avoid malicious interception; at the same time, through ciphertext transmission, it uses the characteristics of holomorphic encryption to perform ciphertext calculation on the server side and feedback the ciphertext results to users. The entire calculation process is fully encrypted to protect data providers' Privacy information; finally, the use of blockchain and innovative contract technology complete the recording of the cloud service computing process and carries out automatic equity distribution to ensure that the service process cannot be tampered with, open and transparent, thereby enhancing the credibility of the service computing model.

***Article Received**: 14 July 2023 **Revised**: 20 August 2023 **Accepted**: 01 September 2023*

A. Trusted Privacy Service Computing Architecture

The trusted privacy service computing architecture is shown in Figure 1, which can be divided into three types of roles: client, model provider, and cloud server, and operates around computing, encryption, and trusted rights and interests.

1) Client

The client is the user of service computing and owns data and public and private key generators. The client needs to make demands on the cloud server, request services, obtain corresponding permissions (such as authentication account opening), and start the entire service process. First, The client generates public and private keys and sends the public key to the cloud server, as shown in Figure 1 ; secondly, the data is encrypted locally with the public key, and the ciphertext is uploaded to the cloud server, as shown in Figure 1 ; Thirdly, the user end obtains the ciphertext calculation result and classification label provided by the cloud server, as shown in Figure 1, and decrypts it locally with the private key to obtain the final result; finally, the user end receives the benefit distribution result of the cloud server, and submit the service fee, as shown in Figure 1.

2) Model provider

First, the model provider needs to obtain the encrypted public key from the cloud server (provided by the client to the cloud server), as shown in Figure 1; secondly, the model provider encrypts the trained prediction model with the public key and provides To the cloud server, it is also necessary to offer classification labels (no encryption is required, the order of each classification in the result vector), as shown in Figure 1; finally, after the calculation service is completed, the model provider obtains the rights and interests distribution results and obtains the corresponding fees, as shown in Figure 1.

3) Cloud server

The cloud server provides powerful computing resources and model services to complete client requests. First, the cloud server receives the user's public key and sends it to the model provider to encrypt the prediction model, as shown in Figure 1. Secondly, it is directly encrypted if the cloud server's prediction model is used. Otherwise, the cloud server receives the model provider's encryption model and the client's encrypted data, as shown in Figure 1, and performs ciphertext encryption. The convolutional neural network calculates and returns the ciphertext result to the client. The privacy service calculation process is completed. In addition, during the calculation process, the cloud server calculates resource usage and service provision and submits the blockchain deposit certificate together with the cloud provider information. , and use the blockchain smart contract to implement the equity calculation model, execute it automatically, and distribute the respective costs and benefits of the cloud server, user, and model provider. Usually, the user pays, and the cloud server and model provider profit.

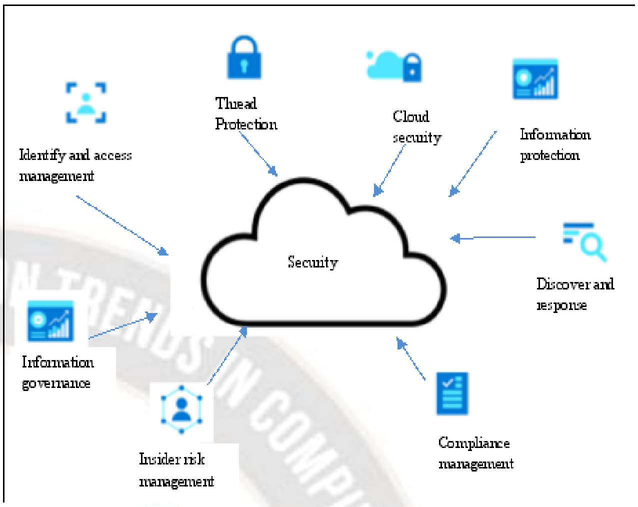

Figure 1. Architecture diagram of trusted privacy service computing

As shown in Figure 1, multiple cloud service providers, in reality, provide different models and services. The data owner has insufficient computing resources, so he must choose a suitable cloud service provider and use its computing power to obtain prediction results while protecting data privacy. The model provider (it can also be a cloud service provider) shares the model and makes profits under the premise of protecting the model content. Homomorphic encryption technology plays a role in protecting data privacy and models in this process. In addition, in this mode, A credible operating environment and equity management mechanism are essential guarantees for breaking monopoly and improving service quality. Blockchain and innovative contract technology can play a role. The use of computing resources, service provision and cloud service provider information are all stored in the blockchain. The system cannot be tampered with, and the rules of intelligent contract calculation rights distribution are transparent, open, automatically executed, and can be queried and held accountable. In addition, the model provider participates in the calculation process and deposits evidence in the blockchain system. This way, clear model ownership can be achieved—the effect of transparent service rights and responsibilities, fair and credible rights and interests. At the same time, there is also a role of a supervisory unit. The supervisory team can view all the certificate data and usage rules to restrain lousy behavior effectively.

B. Prediction service privacy calculation model

In this application scenario, the system and users can choose different holomorphic encryption algorithms according to their needs. The holomorphic encryption in this paper is realized by improving the NDGHV algorithm, which is called the ON-DGHV (Optimized-DGHV) algorithm here. The plaintext of the DGHV algorithm the space is {0, 1} (binary representation). The N-DGHV algorithm transforms the random number multiplied by 2 of the encryption algorithm into multiplied by 2n, and the modulo 2 of the decryption algorithm becomes modulo 2n, thereby expanding the plaintext space from 1 bit to n bit, which reduces the number of encryption times. Further, ONDGHV uses the square public critical compression method to reduce the public key's size and the public key's storage space. However, the model's performance, accuracy and even correctness bring challenges. The model components can be divided into four functional modules, as follows. (1) Homomorphic encryption module: use the public key to encrypt the original data matrix E of the client to obtain D′ , and use the public key to encrypt the model N provided by the model provider to get N` (encrypted model M′ includes convolution Kernel K', convolution offset c1', the fully connected matrix X' and fully connected offset c2').(2) Convolutional neural network prediction module: the encrypted model N′ and the homomorphically encrypted data matrix D′ become the input of the convolutional neural network prediction module. Further, the relationship and function of each level of the convolutional neural network prediction module the operation are shown in Figure 3.The convolution kernel K' and the data matrix E' are used as the input of the convolution layer, and the convolution kernel K' is used to perform convolution calculation on the data matrix E' in the convolution layer to obtain a set of linear output convoy uses the activation function in the activation layer to complete the non-linear mapping operation. To meet the requirements of holomorphic ciphertext, the activation function here is calculated using the square process to generate the ciphertext data acti; then action is summed and pooled through the pooling layer[20], complete data compression, reduce the amount of data, and simplify the calculation complexity, and then output the data pool; finally, put the data pool and the fully connected matrix W' into the fully connected layer for matrix multiplication, and map the features of the upper layer to the sample space To achieve classification, the category with the most considerable median value of all types is the recognition result of CNN, which is expressed as the ciphertext result D. To protect the privacy of the data model, a protection mechanism can be added to the model according to requirements. That is, the ciphertext D The ciphertext state of a random number r is added to each element in (r tries to choose a smaller number and use the same encryption algorithm), the ciphertext of the random number can be expressed as Dr, and after the addition, D (′ is Lock (D) function), that is, D′ = Dr + D, and then send D′ together with the classification labels (the order of each classification in the result vector) to the client. It can be seen from Figure 2 that there is a cascading relationship between the convolutional layer, the activation layer, the pooling layer, and the fully connected layer. The previous layer's output is used as the input of the subsequent layer the calculation of the text data, the effective extraction of data features, and the completion of the prediction function. Decryption module: the user uses the private key to decrypt the obtained ciphertext result D' to obtain U' and gets the classification result corresponding to max(U') according to the classification label, and max(U') is the maximum value in the classification label, that is, the prediction result. Since the encryption algorithm satisfies additive homomorphism and finally judges the classification result according to the numerical value of the elements in the result vector, even if the development of the convolutional neural network prediction module adds Dr random numbers, the final classification result has No effect. However, this will affect the classification probability. Therefore, the cloud can be set according to requirements, and the decryption module also applies to ciphertext D.In this application scenario, the system and users can choose different homomorphic encryption algorithms according to their needs.

The homomorphic encryption in this paper is realized by improving the NDGHV algorithm, which is called the ON-DGHV (Optimized-DGHV) algorithm here. The plaintext of the DGHV algorithm the space is {0, 1} (binary representation). The N-DGHV algorithm transforms the random number multiplied by 2 of the encryption algorithm into multiplied by 2n, and the modulo 2 of the decryption algorithm becomes modulo 2n, thereby expanding the plaintext space from 1 bit to n bit, which reduces the number of encryption times. Further, ONDGHV uses the square public critical compression method to reduce the public key's size and the public key's storage space. However, the model's performance, accuracy and even correctness bring challenges. The model components can be divided into four functional modules, as follows.

(1) Homomorphic encryption module: use the public key to encrypt the original data matrix E of the client to obtain D′ , and use the public key to encrypt the model N provided by the model provider to get N` (encrypted model M′ includes convolution Kernel K', convolution offset c1', the fully connected matrix X' and fully connected offset c2').

(2) Convolutional neural network prediction module: the encrypted model N′ and the homomorphically encrypted data matrix D′ become the input of the convolutional neural network prediction module. Further, the relationship and function of each level of the convolutional neural network prediction module the operation are shown in Figure 3.The convolution kernel K' and the data matrix E' are used as the

input of the convolution layer, and the convolution kernel K' is used to perform convolution calculation on the data matrix E' in the convolution layer to obtain a set of linear output conv uses the activation function in the activation layer to complete the non-linear mapping operation. To meet the requirements of homomorphic ciphertext, the activation function here is calculated using the square process to generate the ciphertext data acti; then action is summed and pooled through the pooling layer[20], complete data compression, reduce the amount of data, and simplify the calculation complexity, and then output the data pool; finally, put the data pool and the fully connected matrix W' into the fully connected layer for matrix multiplication, and map the features of the upper layer to the sample space To achieve classification, the category with the most considerable median value of all types is the recognition result of CNN, which is expressed as the ciphertext result D. To protect the privacy of the data model, a protection mechanism can be added to the model according to requirements. That is, the ciphertext D The ciphertext state of a random number r is added to each element in (r tries to choose a smaller number and use the same encryption algorithm), the ciphertext of the random number can be expressed as Dr, and after the addition, D (′ is Lock (D) function), that is, D′ = Dr + D, and then send D′ together with the classification labels (the order of each classification in the result vector) to the client. It can be seen from Figure 2 that there is a cascading relationship between the convolutional layer, the activation layer, the pooling layer, and the fully connected layer. The previous layer's output is used as the input of the subsequent layer the calculation of the text data, the effective extraction of data features, and the completion of the prediction function.

(3) Decryption module: the user uses the private key to decrypt the obtained ciphertext result D' to obtain U' and gets the classification result corresponding to max(U') according to the classification label, and max(U') is the maximum value in the classification label, that is, the prediction result. Since the encryption algorithm satisfies additive homomorphism and finally judges the classification result according to the numerical value of the elements in the result vector, even if the development of the convolutional neural network prediction module adds Dr random numbers, the final classification result has No effect. However, this will affect the classification probability. Therefore, the cloud can be set according to requirements, and the decryption module also applies to ciphertext D.

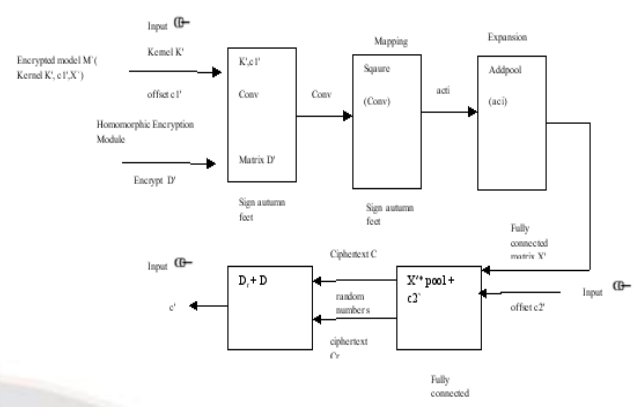

Figure 2. Schematic diagram of the relationship between each level of theconvolutional neural network prediction module

The job of this module is to generate a key pair and encrypt plaintext m. When developing a private key SK, it is necessary to ensure that, where n is the number of bits (bits) of m, and s′ is the encryption time is a random positive integer. The private key SK is a random sizeable prime number. The generated public key set is PK, and the number of elements in the group is 2l+1 (l is a positive integer). The first element in the collection is expressed as dkf = qfSK, where qf is a random positive odd number; it is recommended to take a more significant number. The generation of subsequent elements in the set is described as follows: Randomly generate a positive integer, when is required, effect a random integer in the interval [0,rf), and calculate The key pair generation algorithm is shown in Algorithm 1.

Algorithm 1 Generate key pair

Input: plaintext m, number of bits (bits) n of m, positive integer l (the number of elements contained in the public key is 2l+1)

Output: private key SK, encrypted public key PK

```
1. Function GenKey(n,o,l)
2. generate random positive integer number s′
3. do
4. generate random large prime number y
5. while |n + 2ms′| ≥y/2
6. SK  y
7. generate random positive odd number
8.if
9. put into PK
10. for i from 0 to 1 step 1
11. for j from 0 to k - 1 step 1
12. do
13. generate random positive integer number sij
14. while sij≤ s′
15. do
16. generate random integer number sij
```

```
while sij< 0 or sij ≥rf
while si<j
put pki,j into PK
end for
end for
return SK,PK
end function
```

After obtaining the public key, the following describes the encryption and decryption process by taking the plaintext as a positive integer or zero as an example. The processing method of negative integers is similar, but it needs to be processed according to the specific meaning of the modulo operation in different programming languages. The encryption process is. First, on average, the 2l public keys of the non-initial elements in the PK are divided into two groups. The number of factors in each group is k. Then a public key is randomly selected from each of the two groups of public keys to multiply, and b(0 < b ≤l2) times, and finally add up the results of the times of multiplication to get the sum value, that is, c is a randomly generated positive integer, using PK[X][Y] to represent the public key elements of two groups. At the same time, generate a random positive integer r. Then, calculate the ciphertext is the number of bits in plaintext m, and pdf is the first element of PK). Corresponding to the model described above, the sum is T′ , n is E or N, and c is E’ or N’ .

Algorithm 2 Data Encryption

Input: plaintext m, number of digits n, public key PK, positive integer l (the number of elements contained in the public key is 2l+1)

Output: encrypted ciphertext d

```
Function Encrypt (PK,n,l)
do
generate random positive integer number b
while b > l2
sum 0
for i from 0 to a - 1 step 1
do
generate random positive integer number f1
generate random positive integer number f2
while f1>l or f2>l
generate random positive integer number c
do
end for
generate random positive integer number r
end for a
return d
end function
```

III. Experiment and Analysis

This section conducts an experimental analysis of the privacy service computing model, which is mainly divided into two parts algorithm comparison and ciphertext prediction analysis.

i) Algorithm comparison experiment

The experiment compares the ON-DGHV algorithm used in this paper with the NDGHV algorithm and the CMNT algorithm (1 bit). The three algorithms all use asymmetric homomorphic encryption, and the experiment is carried out from the number of encrypted data and the number of bits.

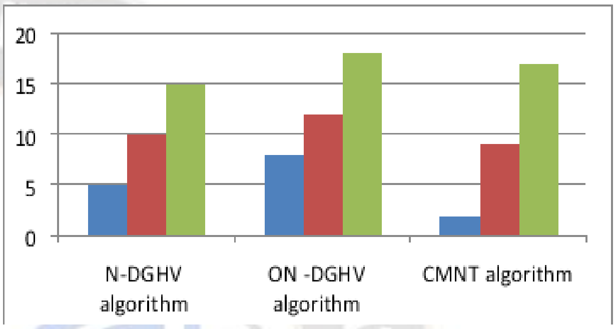

Figure 3. Comparison chart of the changes in the three algorithms

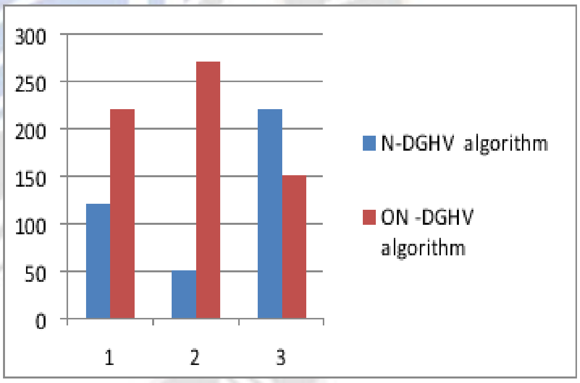

Figure 4 Comparison of decryption time with the change of data bits

The experimental object is 10,000 data, set as a standard decimal number, and the number of data digits is changed to 1, 4, 7 and 10, respectively and decryption time. Ten thousand numbers with different digits (randomly generated data) need to be tested 50 times. The average value of the test indicators is used as the result; see Figure 3 and Figure 4. For clarity, the two figures are equally divided. There are two sub graphs, (a) and (b); sub graph (a) is the change comparison of the three algorithms, and sub graph (b) is the time comparison of the two types of DGHV algorithms. As the number of data bits increases, ON -DGHV and N-DGHV algorithms has slight changes in encryption/decryption time and remain unchanged, and when the abscissa (number of digits) is greater than or equal to 4, the encryption/decryption time is significantly shorter than that of the CMNT algorithm. ONDGHV and N-DGHV algorithm can encrypt/decrypt n-bit data simultaneously (in practical applications, preprocessing is required to obtain n). It can directly perform encryption/decryption processing on decimal numbers (bit). In contrast, the CMNT algorithm can only encrypt/decrypt 1-bit data at a time (only Encryption of 0 and 1 is supported); the decimal data needs to be converted into binary, and then the encryption/decryption operation is performed bit by bit, so when the number of data bits is large, the time-consuming is more prominent. That is to say, the CMNT encryption/decryption time depends on the number of data bits, showing a linear relationship.

ii) Ciphertext prediction analysis experiment

This section is based on the numpy library to implement the convolutional neural network computing model mentioned above, which can be divided into the following eight parts: image encryption, model encryption, convolution layer, activation layer, pooling layer, fully connected layer, plus Random numbers and decryption. The ON-DGHV encryption algorithm is used for experiments. This data set is used to recognize handwritten numbers (image classification), and the size of each picture is 28× 28×1, that is, a grayscale image with a length of 28×28, the number of convolution kernels is 6, and the step size is 1. The model is first trained with 60,000 pictures in the test set. Then the 10,000 sample pictures are For the prediction operation, two types of calculations are completed, the plaintext and the ciphertext prediction (set the encryption bit parameter n that supports the predicted picture value), and then calculate the average of the prediction accuracy of each picture. The plaintext is 98.25%, and the ciphertext is 98.16%; the ciphertext has little impact on image classification accuracy. Then analyze the average execution time ratio of plaintext and ciphertext for each layer of CNN, as shown in Table 1, it can be seen that the calculation of ciphertext has a significant impact on the convolutional layer and the real influence of the connection layer is more prominent.

Table I: Comparison table of execution time ratio of each layer of plain ciphertext CNN

Time ratio	**Convolutional layer**	**Activation layer**	**Pooling layer**
Clear text	39.59%	2.20%	54.79%
Ciphertext	66.49%	5.84%	0.49%

IV. CONCLUSION

This paper studies the CNN-oriented blockchain trusted privacy service computing model, using ON-DGHV homomorphic encryption algorithm and blockchain technology, to strengthen data security, privacy protection and credibility in service computing, with the following characteristics.

1) It provides a set of service computing solutions which improves the contradiction between service computing and data privacy protection. While enjoying the convenience of cloud service computing, it also protects user privacy and security and is conducive to the effective integration of resources and data. , and promote the application and development of new technologies.

2) Explore homomorphic encryption and ciphertext intelligent computing methods, and practice them to seek the application feasibility of privacy service computing.

3) Blockchain and innovative contract technology run through the entire business process to enhance the credibility of services and transactions. Model sharing, service and other methods are stored on the chain, and the rights and interests are evaluated by smart contracts, which can increase the transparency of rules and traceability of rights and responsibilities; rights and interests can be guaranteed, enhancing the practicability of privacy computing services. At the same time, data privacy protection technology can also improve the security of blockchain and smart contracts and expand its application range. This research is still in the initial exploration stage, mainly focusing on the basic homomorphic encryption algorithm, CNN model and data set.

REFERENCES

[1] G. Rao, Y. Zhang, L. Zhang, Q. Cong and Z. Feng, "MGL-CNN: A Hierarchical Posts Representations Model for Identifying Depressed Individuals in Online Forums," in IEEE Access, vol. 8, pp. 32395-32403, 2020, doi: 10.1109/ACCESS.2020.2973737.

[2] G. Shu, W. Liu, X. Zheng and J. Li, "IF-CNN: Image-Aware Inference Framework for CNN With the Collaboration of Mobile Devices and Cloud," in IEEE Access, vol. 6, pp. 68621-68633, 2018, doi: 10.1109/ACCESS.2018.2880196.

[3] M. J. Horry et al., "COVID-19 Detection Through Transfer Learning Using Multimodal Imaging Data," in IEEE Access, vol. 8, pp. 149808-149824, 2020, doi: 10.1109/ACCESS.2020.3016780.

[4] S. Yang, Z. Zhang, C. Zhao, X. Song, S. Guo and H. Li, "CNNPC: End-Edge-Cloud Collaborative CNN Inference With Joint Model Partition and Compression," in IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 12, pp. 4039-4056, 1 Dec. 2022, doi: 10.1109/TPDS.2022.3177782.

[5] D. Kollias and S. Zafeiriou, "Exploiting Multi-CNN Features in CNN-RNN Based Dimensional Emotion Recognition on the OMG in-the-Wild Dataset," in IEEE Transactions on Affective Computing, vol. 12, no. 3, pp. 595-606, 1 July-Sept. 2021, doi: 10.1109/TAFFC.2020.3014171.

[6] Y. Jia et al., "CroApp: A CNN-Based Resource Optimization Approach in Edge Computing Environment," in IEEE

Transactions on Industrial Informatics, vol. 18, no. 9, pp. 6300-6307, Sept. 2022, doi: 10.1109/TII.2022.3154473.

[7] Z. Zhu, G. Han, G. Jia and L. Shu, "Modified DenseNet for Automatic Fabric Defect Detection With Edge Computing for Minimizing Latency," in IEEE Internet of Things Journal, vol. 7, no. 10, pp. 9623-9636, Oct. 2020, doi: 10.1109/JIOT.2020.2983050

[8] Kuralkar, V. P. ., Khampariya, P. ., & Bakre, S. M. . (2023). A Survey on the Investigation and Analysis for a Power System (Micro- Grid) with Stochastic Harmonic Distortion of Multiple Converters. International Journal of Intelligent Systems and Applications in Engineering, 11(3s), 72–84. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2533

[9] F. Liang, W. Yu, X. Liu, D. Griffith and N. Golmie, "Toward Edge-Based Deep Learning in Industrial Internet of Things," in IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4329-4341, May 2020, doi: 10.1109/JIOT.2019.2963635.

[10] F. Liang, W. Yu, X. Liu, D. Griffith and N. Golmie, "Toward Edge-Based Deep Learning in Industrial Internet of Things," in IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4329-4341, May 2020, doi: 10.1109/JIOT.2019.2963635.F. Liang, W. Yu, X. Liu, D. Griffith and N. Golmie, "Toward Edge-Based Deep Learning in Industrial Internet of Things," in IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4329-4341, May 2020, doi: 10.1109/JIOT.2019.2963635.

[11] Z. Xu, F. Yu, Z. Qin, C. Liu and X. Chen, "DiReCtX: Dynamic Resource-Aware CNN Reconfiguration Framework for Real-Time Mobile Applications," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 40, no. 2, pp. 246-259, Feb. 2021, doi: 10.1109/TCAD.2020.2995813.

[12] S. Montaha, S. Azam, A. K. M. R. H. Rafid, M. Z. Hasan, A. Karim and A. Islam, "TimeDistributed-CNN-LSTM: A Hybrid Approach Combining CNN and LSTM to Classify Brain Tumor on 3D MRI Scans Performing Ablation Study," in IEEE Access, vol. 10, pp. 60039-60059, 2022, doi: 10.1109/ACCESS.2022.3179577.

[13] Steven Martin, Thomas Wood, María Fernández, Maria Hernandez, .María García. Machine Learning for Educational Robotics and Programming. Kuwait Journal of Machine Learning, 2(2). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/179

[14] P. Foldesy, L. Kek, A. Zarandy, T. Roska and G. Bartfai, "Fault-tolerant design of analogic CNN templates and algorithms-Part I: The binary output case," in IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 46, no. 2, pp. 312-322, Feb. 1999, doi: 10.1109/81.747209.

[15] L. Ren, J. Dong, X. Wang, Z. Meng, L. Zhao and M. J. Deen, "A Data-Driven Auto-CNN-LSTM Prediction Model for Lithium-Ion Battery Remaining Useful Life," in IEEE Transactions on Industrial Informatics, vol. 17, no. 5, pp. 3478-3487, May 2021, doi: 10.1109/TII.2020.3008223.

[16] L. Ale, N. Zhang, H. Wu, D. Chen and T. Han, "Online Proactive Caching in Mobile Edge Computing Using Bidirectional Deep Recurrent Neural Network," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 5520-5530, June 2019, doi: 10.1109/JIOT.2019.2903245.

[17] A. Kumar, A. Sharma, V. Bharti, A. K. Singh, S. K. Singh and S. Saxena, "MobiHisNet: A Lightweight CNN in Mobile Edge Computing for Histopathological Image Classification," in IEEE Internet of Things Journal, vol. 8, no. 24, pp. 17778-17789, 15 Dec.15, 2021, doi: 10.1109/JIOT.2021.3119520.

[18] T. Roska et al., "The use of CNN models in the subcortical visual pathway," in IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 40, no. 3, pp. 182-195, March 1993, doi: 10.1109/81.222799.

[19] AGYEI , I. T. . (2021). Simulating HRM Technology Operations in Contemporary Retailing . International Journal of New Practices in Management and Engineering, 10(02), 10–14. https://doi.org/10.17762/ijnpme.v10i02.132

[20] T. Roska et al., "The use of CNN models in the subcortical visual pathway," in IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 40, no. 3, pp. 182-195, March 1993, doi: 10.1109/81.222799.

[21] A. H. Al-Badri, N. A. Ismail, K. Al-Dulaimi, A. Rehman, I. Abunadi and S. A. Bahaj, "Hybrid CNN Model for Classification of Rumex Obtusifolius in Grassland," in IEEE Access, vol. 10, pp. 90940-90957, 2022, doi: 10.1109/ACCESS.2022.3200603. Kwame Boateng, Machine Learning in Cybersecurity: Intrusion Detection and Threat Analysis , Machine Learning Applications Conference Proceedings, Vol 3 2023.

[22] T. Li, M. Hua and X. Wu, "A Hybrid CNN-LSTM Model for Forecasting Particulate Matter (PM2.5)," in IEEE Access, vol. 8, pp. 26933-26940, 2020, doi: 10.1109/ACCESS.2020.2971348.

[23] Y. Weng and H. Zhou, "Data Augmentation Computing Model Based on Generative Adversarial Network," in IEEE Access, vol. 7, pp. 64223-64233, 2019, doi: 10.1109/ACCESS.2019.2917207.