




## Article

# An Accelerated Method for Protecting Data Privacy in Financial Scenarios Based on Linear Operation

Huairong Huo <sup>1,†</sup> , Jiangyi Guo <sup>2,†</sup>, Xinze Yang <sup>2,†</sup> , Xinai Lu <sup>3,4,†</sup>, Xiaotong Wu <sup>2,†</sup>, Zongrui Li <sup>2,†</sup> ,  
Manzhou Li <sup>5</sup> and Jinzheng Ren <sup>2,\*</sup>

<sup>1</sup> College of Humanities and Development Studies, China Agricultural University, Beijing 100083, China

<sup>2</sup> College of Economics and Management, China Agricultural University, Beijing 100083, China

<sup>3</sup> International College Beijing, China Agricultural University, Beijing 100083, China

<sup>4</sup> Economics, University of Colorado Denver, Denver, CO 80202, USA

<sup>5</sup> College of Plant Protection, China Agricultural University, Beijing 100083, China

\* Correspondence: rjzheng@cau.edu.cn

† These authors contributed equally to this work.

**Abstract:** With the support of cloud computing technology, it is easier for financial institutions to obtain more key information about the whole industry chain. However, the massive use of financial data has many potential risks. In order to better cope with this dilemma and better protect the financial privacy of users, we propose a privacy protection model based on cloud computing. The model provides four levels of privacy protection according to the actual needs of users. At the highest level of protection, the server could not access any information about the user and the raw data, nor could it recover the computational characteristics of the data. In addition, due to the universality of the mathematical principle of linear operators, the model could effectively protect and accelerate all models based on linear operations. The final results showed that the method can increase the speed by 10 times, compared with the privacy protection method that only uses local computing power instead of the cloud server. It can also effectively prevent the user's privacy from being leaked with relatively minimal delay cost, compared with no privacy protection method. Finally, we design a multi-user scheduling model to deploy the model in a real scenario, which could maximise server power and protect user privacy as well.

**Keywords:** privacy protection; cloud computing; financial scenarios; linear operation; encryption and decryption



**Citation:** Huo, H.; Guo, J.; Yang, X.; Lu, X.; Wu, X.; Li, Z.; Li, M.; Ren, J. An Accelerated Method for Protecting Data Privacy in Financial Scenarios Based on Linear Operation. *Appl. Sci.* **2023**, *13*, 1764. <https://doi.org/10.3390/app13031764>

Academic Editor: Dimitris Mourtzis

Received: 2 December 2022

Revised: 4 January 2023

Accepted: 5 January 2023

Published: 30 January 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In finance, a data-intensive industry, data is highly sensitive and valuable. Comprehensive protection of data and ownership of personal information has become important propositions in the use and management of data in the financial industry [1].

Nowadays, with the conglomeration of technology and finance, financial institutions tend to use big data technology to collect, store, and correlate and analyse massive amounts of financial data from scattered sources and in various formats, to draw useful insights from them and continuously upgrade the functions of financial services in the industry. In other words, financial institutions can use big data technology to further map the overall situation of enterprises, combine information concerning the industry growth, historical development status, and regional economic characteristics, and clarify the direction of the economy. On this basis, finance within the industrial chain can be upgraded through innovative financial products and optimised credit approval models to help enterprises solve their realistic financing dilemmas. At the same time, the enterprise's actual sales can be inferred by examining information such as tax payment, inventory, and capital turnover in multiple dimensions through big data technology, and its existing sales data can be reviewed and verified through analysis of the customer group's account capital

stock, financing needs, and risk reality. This can effectively help quality enterprises to better identify fraud, money laundering, and other risks while improving the financial support services of the industry chain [2,3].

At the same time, the exponential growth of data volume is very important for the bank and financial industry, improving the security of financial information data [4–6]. As financial systems continue to increase and upgrade, more and more financial entities are interconnected in various ways, which also poses a huge challenge to information security regulation. Financial information is exposed to various potential threats, which may lead to serious consequences if financial information security is not well safeguarded [7,8].

In addition, the application of cloud computing can effectively alleviate the load on the financial participants and enterprise groups in the industry chain, which are trying to build their information systems [9,10]. At present, the upstream and downstream of the industry chain are mostly small and medium-sized enterprises. These enterprises generally have limited technical investment, and their information systems only support regular business operations and cannot match the need of financial institutions and financial technology enterprises. Under such circumstances, on the one hand, the enterprises in the chain cannot enjoy the dividends brought by supply chain finance, and on the other hand, the financial institutions will face the challenge of great difficulty in docking in developing and carrying out supply chain finance. Cloud computing breaks through the constraints of hardware and software, and is capable of allowing access by demand, which can solve the pressure of constructing and docking the information system. To a certain extent, this technology can meet the requirements of storage, calculation, and analysis of financial information in the industry chain, and significantly improve the digitalisation of the industry chain's financial business. In terms of practical application, with the support of cloud computing technology, financial institutions can cross-analyse key information such as the financial situation of enterprises in the whole industrial chain and the competitive relationship between enterprises, from multiple sources and in multiple dimensions, and make full use of them, so that all enterprises in the chain can share the dividends of supply chain finance [11]. Consequently, fin-tech, represented by big data technology and cloud computing applications, has played a significant role in the development of the financial industry. However, the massive use of financial data also contains a large number of potential risks.

At the macro level, improper use or leakage of personal or financial institution data will not only directly infringe upon the legitimate rights and interests of individual financial information subjects and affect the normal operation of financial industry institutions, but may also bring about systemic financial risks and threaten financial security, and in serious cases, may spread to the whole society along with the economic chain. At the same time, the occurrence of information infringement and other violations of the law by micro-entities makes it difficult for citizens to complain, and the failure to defend their rights brings huge economic and psychological damage to citizens, which may lead to incidents driven by irrational behaviour, affecting social stability and causing certain damage to the government's credibility.

At a micro level, it infringes on the interests of consumers and financial institutions [12,13]. For consumers, firstly, there is a risk of data misuse and privacy breaches. With the rapid development of digital consumption, financial consumers using financial applications or third-party software for financial services are mandatorily required to allow the right to collect or query information in privacy agreements or various authorisations. All online information of users including identity, location, shopping preferences, payment passwords, and other types of information are recorded in the background. It is not uncommon for financial data to be misused and leaked, by means like over-marketing and big data discriminatory pricing. For financial institutions, firstly, there is the risk of data leakage. Insiders may be bribed to illegally sell or deliberately leak information or be supported by hackers to cause data leakage, which results in losses due to the market going less than expected. Secondly, there is the risk of data contamination. The unlabelled nature of data is

highly susceptible to copying and tampering in the course of digital transactions, and once the sample is maliciously damaged, the model results will be very different, which will cause high data clean-up costs for financial institutions, and affect the institutional decision making [14].

Based on the above analysis, we can find that although there are many methods for data encryption and privacy protection, these methods have one or more of the following drawbacks:

1. It is based on a specific hardware, such as TEE, which makes the method too narrow in scope.
2. The computational complexity of the encryption algorithm is too high, and the computational overhead of encryption is too large.
3. It is not optimised for financial computing scenarios.

Many algorithms in the field of finance, such as Monte Carlo algorithms under linear models, n-dimensional discontinuous segmented linear financial market models, financial time series analysis, etc., are linear in their core computation, i.e., matrix multiplication [15–21]. These computationally intensive operations are suitable for computing in the cloud (e.g., cloud GPUs) [22].

This paper, therefore, proposes a security model for distributed applications that can secure data even if an attacker has physical access to the cloud server. It provides four levels of privacy protection. In the highest level of protection, the server cannot access any information about the user of the data, nor the original text of the data, nor the computational characteristics of the data, such as computational weights and gradients, nor the statistical characteristics of the data, such as the data distribution. Moreover, due to the generality of linear computing in mathematical principles, this model can effectively protect and accelerate all linear computing-based models. It has been experimentally validated that this approach can improve the inference speed of algorithms by up to 10 times compared to a benchmark test using the only client-side computing power without compromising privacy, and can effectively prevent the restoration of user data compared to accelerated operations without privacy protection.

In summary, in this paper, we design a privacy-preserving computational framework to address the specific computational properties of financial scenarios, and our main contributions are as follows:

1. This paper proposes a computing framework based on client-side encryption and decryption, accelerated computing in the cloud.
2. This paper adopts this framework to many algorithms in the field of finance, such as Monte Carlo algorithm under linear model, n-dimensional discontinuous linear financial market model, and financial time-series analysis.
3. This paper designs a scheduling model for multi-client shared cloud GPU.
4. This paper has implemented a large number of experiments to verify the effectiveness of this method.

## 2. Related Work

Nowadays, blockchain technology is being used within the field of financial security. Blockchain is a P2P (peer-to-peer) distributed database and is a web-based concept. It consists of a series of blocks containing transactions, timestamped and verified by the network community and protected by a PKI (Public Key Infrastructure). An element of the blockchain cannot be modified after it has been added to the blockchain. Moreover, it retains a permanent account of earlier actions [23]. Furthermore, blockchain is expected to initiate an industrial and commercial revolution, while contributing to global economic reform. First, blockchain uses cryptography to create a secure code in digital form. Users can then confirm purchases without having to provide any personal data. Since blockchain records are immutable, transactions are automatically completed and decentralised [24]. The blockchain is a secure database and decentralised transaction system driven by decen-

tralised nodes [25]. In other words, blockchain is a game-changing technology that has attracted the attention of businesses and governments worldwide. Essentially, the term “distributed ledger technology” refers to the collection of transactions and data that are sequentially tracked and registered on a network of distributed ledgers [26]. A blockchain is also divided into an ever-expanding list of records, called blocks, linked together using cryptography [27]. Blockchains are used as transaction ledgers in cryptocurrency systems such as Bitcoin and Ether, where the blockchain records the current state and previous transactions. Blockchains can additionally be defined as collections of blocks that store data in a hash function and include a link to the previous block and timestamp [28]. A blockchain is a distributed database that only allows new data to be appended to existing data [29]. Based on this feature, blockchain technology plays an important role in the financial sector. The first generation of blockchain technologies acted as currencies, such as Bitcoin. In addition to Bitcoin, the most prominent cryptocurrency, about 600 other cryptocurrencies have been established and are used as exchange tokens in Bitcoin-based applications. Ether, Monero, and Ripple are the other most popular ones [30]. The second generation of blockchain technology is not only about cryptocurrency transactions but also about bonds, smart contracts, futures, loans, and mortgages. The integration of smart contracts with the blockchain is the most critical feature of this phase. When certain criteria are met, smart contracts are parts of code buried within them that react in a specific way [30]. Furthermore, blockchain can solve the problem of e-commerce reputation schemes that use the registration of a large number of fake customers to gain a high reputation; the reputation information is not reversible because it is stored on the blockchain and all reputation changes are easily detectable. For security enhancements, blockchain can solve the problem of a single point of failure of an important central node. And since it can reduce the impact of attacks on public and private key distribution devices, Blockchain can help build more reliable public–private key infrastructures [31]. Blockchain technology can also be used for privacy protection in the financial sector, where a data storage system built on blockchain can guarantee the anonymity of users while ensuring their ownership of data [32].

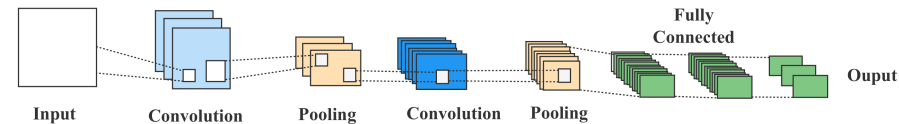
In addition, cloud computing is being widely used in enterprises. For large enterprises and governments, much of the appeal of the cloud lies in better control and reduction of data centre costs. In particular, cloud computing allows the reclassification of IT expenses from capital expenditure to cash-based ‘operational expenditure’ [33]. Another business benefit of cloud computing is reduced demand for skilled labor in places where there is a shortage of high technology (e.g., South East Asia) [34]. Moreover, it lowers the barrier to entry into computing [35–37]. However, with its widespread use, cloud computing also poses security concerns. A key challenge in cloud computing security relates to the environment in which the computing takes place. Cloud computing providers operating globally distributed networks of data centres may face specific security risks (e.g., terrorism or cyber-attacks) and may also have unique legal issues regarding security tort liability [38,39]. Certain user environments have specific security requirements, such as governments and financial institutions [40]. They may ask providers to promote the security level.

Furthermore, several studies have been conducted to improve data security by desensitising the data itself. Tuple repetition-based privacy-preserving data publishing schemes have been proposed to hide sensitive information from individuals when publishing data. It is effective in addressing the issue of releasing data collected by the data holders or publishers from the data owners, which means it is capable of protecting sensitive information about an individual’s released data. Anonymisation techniques such as generalisation, suppression, swapping, disaggregation, and randomisation can suffer from loss of personal identity or other information, which can reduce the usefulness of the data [41]. Moreover, one study proposes a framework that allows the use of random data perturbation techniques, systematically transforming the original data and presenting the modified data to parties as query results through a decision tree. This approach provides valid results for analytical purposes, but the actual or real data is not revealed and privacy is protected [42].

### 3. Proposed Methods

#### 3.1. Overall

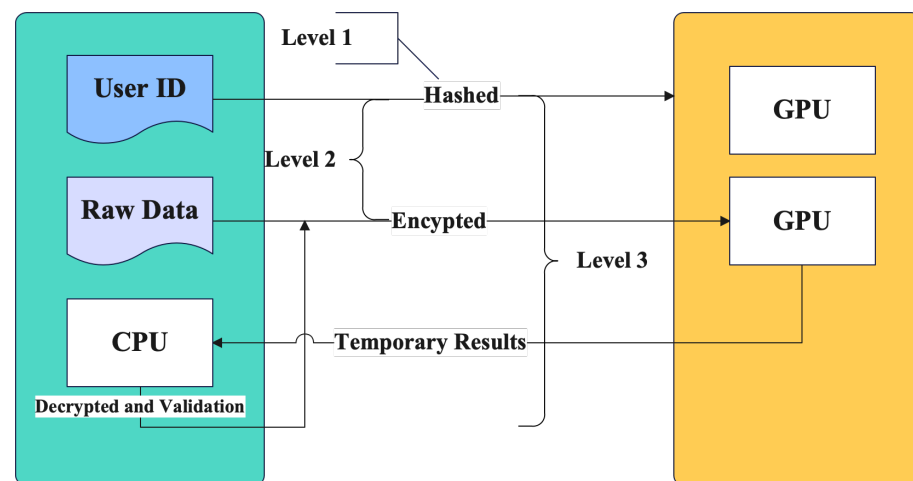
A prediction algorithm or computational model, such as the representative deep neural network (DNN) model with a large variety of operations in circulation so far, can be represented as a sequence of layers, each layer containing a set of operators that are either linear or nonlinear, as shown in Figure 1.



**Figure 1.** Illustration of a common DNN model.

The above diagram shows the inference process of the model, which puts  $X$  into the model  $M$ , computed by the operation layer, and finally outputted. In the case of a DNN network, the first step is convolution, followed by activation using the activation function, and so on, and the final result is obtained by probabilistic output using the softmax function.

From the above analysis, it can be seen that linear operations have been shown to occupy a large portion of the arithmetic resources in many inferences and prediction algorithms and models, without loss of generality. Therefore, in our model, we transmit most of the linear operators in the model to the cloud for computation, while a small number of non-linear operators and encryption and decryption in the model that do not require much computational power are computed locally by the user. The overall system design is shown in Figure 2 and the detail will be introduced in Section 3.2.



**Figure 2.** Illustration of protection levels supported by proposed method.

#### 3.2. Level of Protection and Attacker Assumptions

First, we define the following four protection levels that can be provided by this method:

1. Level 0: No privacy protection. This level of protection means that user data will be transmitted and calculated in plaintext form.
2. Level 1: User identity is protected. This level of protection means that this file method applies hash processing to the user's identity to prevent explicit data files from revealing information about the data owner in the real world. Protection Effectiveness: The effect of this protection is similar to the way that the Internet advertising systems of companies such as Countrywide add noise to the user's identity when using their data to make it obscure and irreversible.
3. Level 2: Protects original data files. This level of protection prevents the original data that the user needs to compute from leaving the user's side and replaces it with an encrypted file.

4. Level 3: Protects the computational and statistical characteristics of data created during model computation. This model is the highest level of privacy protection. In this mode, the server cannot obtain any information about the data user, nor can it obtain the original file of the data, nor can it restore the computational characteristics of the data, such as computational weights and gradients, or the statistical characteristics of the data, such as data distribution.

These four protection levels correspond to the five data processing paths in Figure 2.

Since these four levels of protection are lower and included in each class, i.e., Level 3 includes all functions from Level 0 to Level 2, this document only describes the implementation details of this model using Level 3 as an example.

### 3.2.1. User Identity Protection

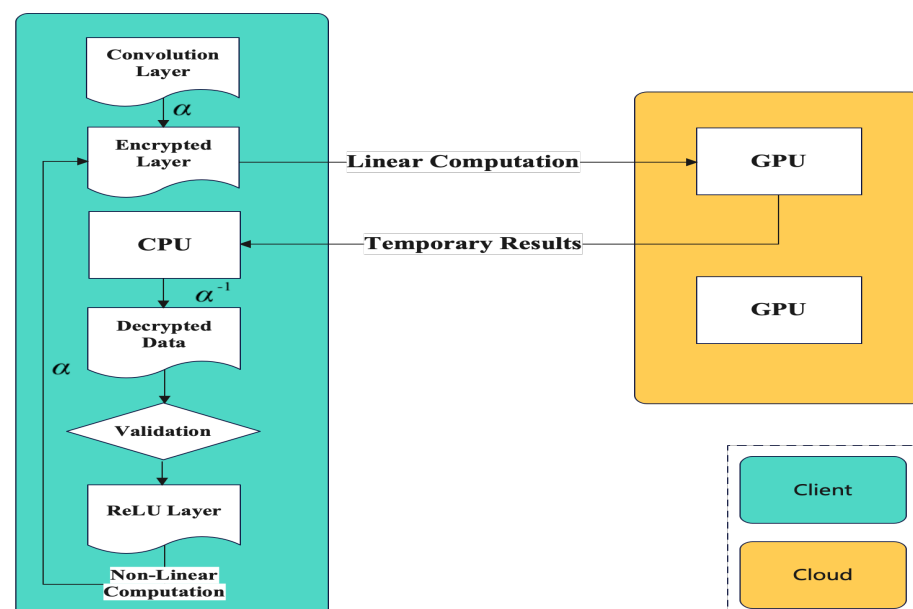
As seen in Figure 2, this method uses a hash algorithm to process user ids and user device IDs. Since hash computation is irreversible, it can effectively protect the privacy of the user's identity.

### 3.2.2. Raw Data Protection

In this section, we describe the implementation of encrypted data, assuming input  $X$ , a vector  $\alpha$ , and a linear operator  $M$ , satisfying Equation (1).

$$(\alpha^{-1} \times \alpha) \times F(X) = \alpha^{-1} \times F(\alpha \times X) \quad (1)$$

Take the convolutional algorithm commonly used in the DNN network as an example. If each weight in the convolution kernel  $R$  is multiplied by  $\alpha$ , which is a scalar, the output is  $\alpha R$ . It can be seen that it is consistent with Equation (1). In this case, we use Equation (1) as the  $(\alpha^{-1} \times \alpha) \times R = \alpha^{-1} \times (\alpha R)$ . This property is also applicable to other linear operators. We use Equation (1) as the central principle for parameter conversion, to protect user privacy while using the GPU for acceleration. The overall system flow is shown in Figure 2. The overall system flow is shown in Figure 3.



**Figure 3.** Details about implement of the protection, including encryption, decryption, and validation.

As can be seen from the figure, we first encrypt the user file before sending it to the GPU to protect the privacy of the user file.



### 3.2.3. Computational and Statistical Characteristics of the Protection Data Created during Computing

As seen in Figure 2, at level 3, each time the server completes the linear operator at that stage, which is the next nonlinear operator, the intermediate result is sent to the client, which decrypts it and performs the nonlinear operator, and then sends the result to the server after it has encrypted it again so that the server can complete the subsequent linear operation. This process is repeated until the final result is calculated.

### 3.3. Model Validation

In this section, we verify the computational integrity, privacy protection, and communication achievability of the model.

#### 3.3.1. Integrity of Computation

To protect the invisibility of operations, this file does not use fully homomorphic encryption because the complexity of the algorithm is unacceptable and there is a challenge in implementing the proposed method to confirm that the intermediate results returned by the server are correct. It is not possible to verify this by performing operations locally, which would make cloud acceleration meaningless. Therefore, we use the Freivalds algorithm [43] for verification. If it passes, we continue the process until the final result is obtained, and if it fails, we consider the current server to be under attack and stop communication. By using the above algorithm and process, we can verify the integrity of this file.

#### 3.3.2. Protection of Raw Data

To protect the privacy of client data, we use a scalar  $\alpha$  to encrypt each transmitted data, while the corresponding decryption scalar  $\alpha^{-1}$  is stored only on the client side. This method of encryption necessarily results in some additional overhead:

1. We have encryption and decryption overhead.
2. We have transmitted data overhead.
3. We have validation intermediate result overhead.

However, since the server only computes the encrypted linear operations, it needs to communicate with the client when non-linear operations are involved, return the intermediate results to the client for decryption, and then encrypt the new intermediate results back to the server. As a result, the server has no access to the original file, which protects the privacy of the user's data. It is important to recognise that these overheads can cause delays in the creation of this file, which we will discuss in Section 5.1.

### 3.4. Experiment Settings

#### 3.4.1. Platform

We conducted experiments on Intel CPUs (1065G7) as the client and NVIDIA RTX 3080 (as the server-side gas pedal) and tested DNN network models with large amounts of linear computation, including the VGG series [44], the ResNet series [45], and the MobileNet [46]. We use Windows 10 as the operating system and Python 3.9 and PyTorch 1.8 as the programming platform.

#### 3.4.2. Models and Algorithms for Experiment

As mentioned above, we use VGG, ResNet, and MobileNet as the test models. Linear regression or other prediction models are not used because they contain very few operators and therefore cannot simulate the large number of computational requests in real scenarios. Here, we briefly introduce these three models:

1. **VGG.** VGG was proposed by the group of Visual Geometry Group at Oxford. The network is a related work at ILSVRC 2014, and the main work is to demonstrate that increasing the depth of the network can affect the final performance of the network to some extent. There are several structures of VGG, namely, VGG13, VGG16, and

VGG19, which are not fundamentally different from each other, but the network depth is not the same.

One improvement of VGG16 over AlexNet [47] is the use of several consecutive  $3 \times 3$  convolutional kernels instead of the larger convolutional kernels in AlexNet ( $11 \times 11$ ,  $7 \times 7$ ,  $5 \times 5$ ). For a given perceptual field (the local size of the input picture with respect to the output), the use of stacked small convolutional kernels is preferable to the use of large convolutional kernels, because the multilayer nonlinear layers can increase the network depth to ensure learning more complex patterns, and at a smaller cost (fewer parameters). In brief, in VGG, three  $3 \times 3$  convolutional kernels are used instead of  $7 \times 7$  convolutional kernels, and two  $3 \times 3$  convolutional kernels are used instead of  $5 \times 5$  convolutional kernels. The main purpose of this is to improve the depth of the network while guaranteeing to have the same perceptual field, which improves the neural network to some extent. For example, the superposition of one layer of three  $3 \times 3$  convolutional kernels with step size 1 can be viewed as a perceptual field of size 7 (in fact, it means that three  $3 \times 3$  successive convolutions are equivalent to one  $7 \times 7$  convolution), whose total parameters are  $3 \times (9 \times C^2)$ , and if the  $7 \times 7$  convolutional kernels are used directly, the total parameters are  $49 \times C^2$ , where  $C$  refers to the number of channels of input and output. Obviously,  $27 \times C^2$  is less than  $49 \times C^2$ , i.e., fewer parameters; and the  $3 \times 3$  convolution kernel facilitates better preservation of image properties.

2. **ResNet.** The ResNet network is a reference to the VGG19 network with modifications based on it and the inclusion of residual units through a short-circuiting mechanism. The changes are mainly reflected in that ResNet directly uses the convolution with *stride* = 2 for downsampling and replaces the fully connected layer with the global average pool layer. An important design principle of ResNet is that the number of feature maps is doubled when the feature map size is reduced by half, which maintains the complexity of the network layers. ResNet adds a short-circuiting mechanism between every two layers compared to the normal network, which results in residual learning. For the 18-layer and 34-layer ResNet, it performs residual learning between two layers, and when the network is deeper, it performs residual learning between three layers, and the three convolutional kernels are  $1 \times 1$ ,  $3 \times 3$ , and  $1 \times 1$ , respectively, and a noteworthy point is that the number of feature maps in the hidden layer is relatively small and is  $\frac{1}{4}$  of the output. The number of feature maps is  $\frac{1}{4}$  of the number of output maps.
3. **MobileNet.** The basic unit of MobileNet is the depthwise separable convolution, a structure that has been used in the Inception model before. Depthwise convolution is actually a factorised convolution that can be decomposed into two smaller operations: depthwise convolution and pointwise convolution. Depthwise convolution is different from standard convolution in that the convolution kernel is used for all input channels, while depthwise convolution uses different convolution kernels for each input channel, that is, one convolution kernel corresponds to one input channel, so depthwise convolution is a depth-level operation. The pointwise convolution is actually a normal convolution, except that it uses a  $1 \times 1$  convolution kernel. For depthwise separable convolution, the depthwise convolution is used to convolve the different input channels separately, and then the pointwise convolution is used to combine the above outputs, which is actually similar to a standard convolution. The overall effect is similar to that of a standard convolution, but the computational effort and the number of model parameters are greatly reduced. The main purpose of using this network is to test the performance of this method in scenarios with low linearity.

The above three models are used to test the execution efficiency of the methods in this paper. Another encryption algorithm that can be compared with this method is homomorphic encryption, which provides the same level of security protection.

With the help of homomorphic encryption (HE), operating directly on the ciphertext is the same as operating on the plaintext and then encrypting it. A typical application



scenario is that the data holder wants to perform computation on the large amount of data it holds, but the computing resources it has are not enough and it wants to use the computing power of the cloud server to complete the computation. If the data is transferred directly to the cloud server, then a pre-written program is run to perform the computation. But then, the sensitive data is exposed on the cloud server. Homomorphic encryption solves this problem by encrypting the data before the data holder transmits it, and the cloud server computes the data as usual after receiving it, except that this time it is done on the ciphertext, and the cloud server cannot access any original information. When the result is obtained, the ciphertext of the result is returned to the data holder, and the final result is obtained when the data holder decrypts it.

### 3.4.3. Experiment Metric

In this paper, we use the following three evaluation metrics: Accuracy, Precision, and Recall. First, we introduce the confusion matrix. A confusion matrix is a table often used in data science and machine learning to summarise the prediction results of a classification model, represented by a matrix with  $n$  rows and  $n$  columns, which summarises the records in a dataset according to two criteria: the true category and the predicted category. As an example, the confusion matrix for the dichotomous classification task is shown in Table 1.

**Table 1.** Matrix of classification metrics.

Label/Prediction	Positive	Negative
Positive	TP	FP
Negative	FN	TN

TP denotes the number of samples that are positive but predicted to be positive, FP denotes the number of samples that are negative but predicted to be positive, FN denotes the number of samples that are positive but predicted to be negative, and TN denotes the number of samples that are negative but predicted to be negative.

Based on the above discussion, the formula for accuracy can be derived:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Accuracy is the simplest and most intuitive evaluation metric in classification problems, but there are obvious drawbacks. For example, if 99% of the samples are positive, then the classifier only needs to predict positive all the time to get 99% accuracy, but its actual performance is very low. That is, when the proportion of samples from different categories is very unbalanced, the category with a large proportion tends to be the most important factor affecting the accuracy. Therefore, this paper introduces the evaluation index of accuracy rate, which is calculated as:

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Both of these metrics are concerned with the proportion of samples that are correctly predicted by the model in the model prediction or the total. The recall is the proportion of actual positive samples that are predicted to be positive out of the actual positive samples and is calculated as:

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

The recall is intuitively the ability of a classifier to find all positive samples. The best value of recall is 1 and the worst value is 0.

## 4. Results

### 4.1. Verification Results

In this section, we validate the inference models mentioned in Section 3.4, as shown in Table 2. The results are the execution time tests using inference from the ImageNet dataset.

**Table 2.** Execution time for different methods.

Inference Model	Local CPU	HE	Our Method
VGG 13	4.3 min	6.5 min	43.1 s
VGG 16	6.1 min	7.3 min	44.8 s
VGG 19	7.4 min	10.5 min	46.5 s
ResNet 50	4.4 min	5.4 min	37.3 s
ResNet 101	10.7 min	11.8 min	54.2 s
MobileNet	4.1 min	5.7 min	35.6 s

From the above table, we can see that the computational efficiency of this method can be improved by a factor of 10 compared to the same security level of the homomorphic implementation or the scenario where the computation is performed exclusively on the local CPU. This is because this method can accelerate a large amount of linear computation using the GPU compared to a local CPU-only scenario, whereas this method can significantly improve the efficiency of encryption compared to homomorphic encryption. Since this method distinguishes linear operations from nonlinear operations for the characteristics of inference networks, we encrypt linear operations that consume large amounts of computing power and send them to the GPU for acceleration, while nonlinear operations that do not consume computing power and are not easily encrypted are performed in a local trusted environment. In this way, we use both methods to protect the privacy of both operations, which protects the privacy of user data throughout the entire process and makes the entire encryption and decryption process fast and efficient.

### 4.2. Precision Quantisation

Due to the way floating point numbers are stored in the computer, there is an inevitable loss of precision in the encryption and decryption process, which accumulates over time as the number of calculations increases. Therefore, this section discusses this possible loss of model accuracy due to the computation of the encryption and decryption process. In this section, we compare the inference accuracy of the inference model in the non-privacy-protected environment with that of the inference model using this method. The results are shown in Tables 3–5.

**Table 3.** Precision quantisation on our method.

Inference Model	Without Protection	With Our Method
VGG 13	83.6%	83.6%
VGG 16	85.7%	85.3%
VGG 19	86.3%	86.1%
ResNet 50	86.5%	86.6%
ResNet 101	85.8%	85.8%
MobileNet	73.4%	73.1%

As can be seen from the figure, although the accuracy of the model is reduced in some models, the reduction is very low at 0.4%. Even in the ResNet series, this method results in a slight improvement in model accuracy. This result was observed on ResNet 50, an inference model with 50 linear computation layers, and this number of computation layers did not lead to any reduction in inference accuracy. This result shows that although this method accumulates accuracy errors in the computation process, these errors do not necessarily affect the final results. Therefore, it can be concluded that this method has little impact on the final progress of the inference model.

**Table 4.** Accuracy quantization on our method.

Inference Model	Without Protection	With Our Method
VGG 13	81.5%	81.3%
VGG 16	83.9%	83.9%
VGG 19	86.2%	85.9%
ResNet 50	84.3%	86.1%
ResNet 101	84.6%	84.7%
MobileNet	71.9%	72.3%

**Table 5.** Recall quantisation on our method.

Inference Model	Without Protection	With Our Method
VGG 13	78.6%	77.9%
VGG 16	81.5%	80.9%
VGG 19	82.8%	82.9%
ResNet 50	83.7%	83.7%
ResNet 101	83.8%	83.8%
MobileNet	66.3%	61.8%

## 5. Discussion

### 5.1. Model Acceleration

#### 5.1.1. Latency Quantisation

To accelerate the model effectively, we first quantified the latency of this method on the ResNet 101 model, which has the most significant number of computational layers, as shown in Table 6.

**Table 6.** Details about latency distribution.

Encryption and Decryption	Results Validation	Transmission
7.8 s	3.4 s	23.0 s

As can be seen from the table, the main delay of the model is as follows:

1. The model validates the intermediate results of the calculation several times to verify the validity of the intermediate results calculated by the server.
2. The model is a single-threaded mechanism in the process of network transmission and server computation, in which the computation power is not effectively used, i.e., the server computation power waits for data transmission.
3. Due to the nonlinear operator of the tested algorithm, a large amount of communication between the client and the server is created.

Based on the above experimental results and analysis, three optimisation methods were designed: pre-calculation of some validation values; batch transmission of data; and modification of the calculation model.

#### 5.1.2. Precalculation

Since the parameters of the algorithm or model used by the user are fixed when it is deployed on the server, we can reduce the model latency by storing some of the computations locally before deployment and directly comparing them with each other during the inference process. The results of our experiments are shown in Table 7.

As can be seen from the above table, when using the precomputed optimisation method, the latency at the stage of verifying intermediate results can be reduced by 74%.

**Table 7.** Comparison of execution time by real-time computation and precomputation.

Inference Model	Real-Time Computation	Precomputation
VGG 13	0.8 s	0.5 s
VGG 16	0.8 s	0.5 s
VGG 19	0.8 s	0.6 s
ResNet 50	2.7 s	0.7 s
ResNet 101	3.4 s	0.9 s
MobileNet	0.2 s	0.1 s

### 5.1.3. Batch Processing

As shown in Table 6, since the communication cycle is often longer than the server's computation cycle, the server's computation power can be improved by combining batches to reduce model latency. The user's data is encrypted and packaged for transmission as a batch stream so that one batch is computed on the server while multiple batches are transmitted, thus alleviating the mismatch between transmission and computation cycles. The comparison between before and after optimisation is shown in Table 8.

**Table 8.** Comparison of execution time by different processing.

Metirc	Model	Result
Number of transmission	single processing	76
	batch processing	17
Time of transmission	single processing	23.0 s
	batch processing	14.1 s

As shown in the table above, batch processing can greatly increase the degree of parallelism in the execution of this method, thereby increasing the utilisation of the GPU computation core and reducing the impact of communication, ultimately reducing the latency of the communication phase by 38.6%.

### 5.1.4. Pure Linear Operator Model

From Table 6 and Section 5.1.3, it can be seen that reducing the communication cost can effectively reduce the model latency; Section 5.1.3 uses parallelisation to reduce the loss of a single communication, instead of reducing the number of communications to reduce the model latency in this section. We modify the operator of the test algorithm by replacing the nonlinear operator with a linear operator. The experimental results are shown in Table 9.

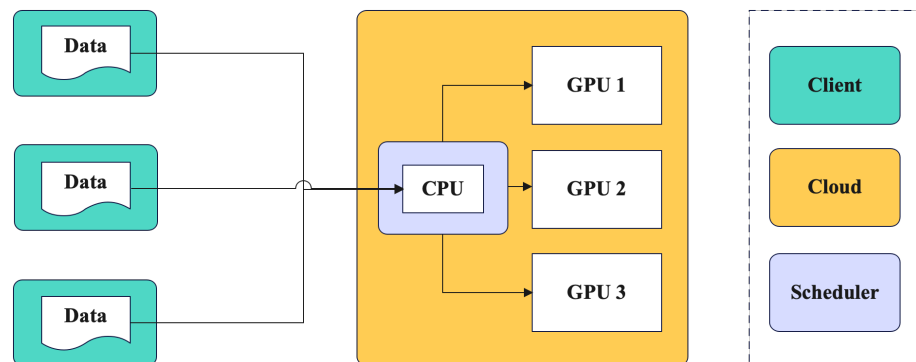
**Table 9.** Comparison of execution time on different model.

Metirc	Model	Result
Number of transmission	original model	76
	remove linear layer	31
Time of transmission	original model	23.0 s
	remove linear layer	12.8 s

From the above table, it can be seen that by replacing the nonlinear arithmetic, the number of communications can be greatly reduced, and the delay of this method can be reduced by 44.3%.

## 5.2. Multi-User Scheduling Model

To deploy this method in a cloud environment and make it available to multiple users, we have designed a multi-user scheduling model as shown in Figure 4.

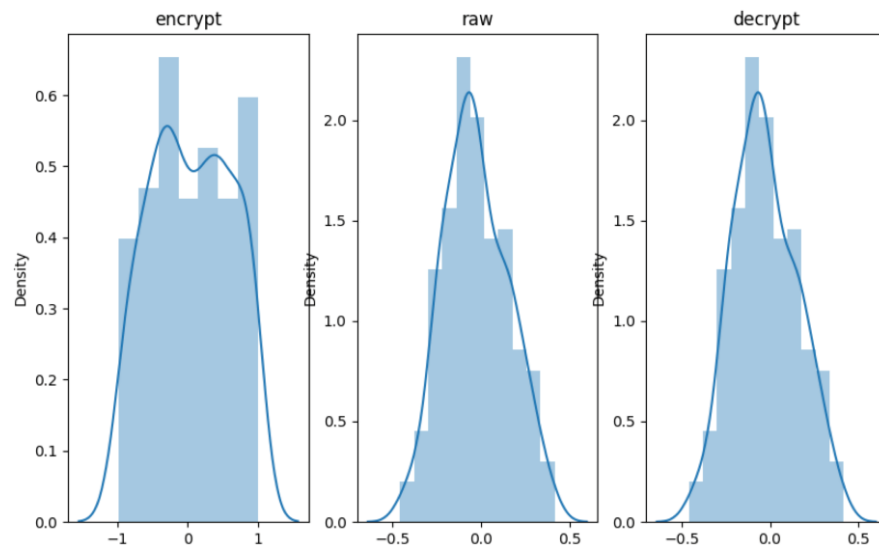


**Figure 4.** Illustration of the scheduling model.

In order to accommodate the different scenarios of multi-task usage by multiple users, i.e., each user has different urgency levels for different tasks, the scheduling model schedules the user's tasks according to these values.

### 5.3. Visualisation of Encryption Effects and Robustness Validation

In this section, to more visually discuss the impact of our approach on computational data privacy protection, we visualise the original data and its encrypted data distribution as shown in Figure 5. In addition, to verify the robustness of our approach, the dataset used in this section is the negative financial information and subject determination dataset provided by DataFountain. The results are shown in Table 10.



**Figure 5.** Visualisation of data distribution based on the encryption and decryption of raw data by the method in this paper.

**Table 10.** Accuracy quantisation on our method with named entity recognition dataset.

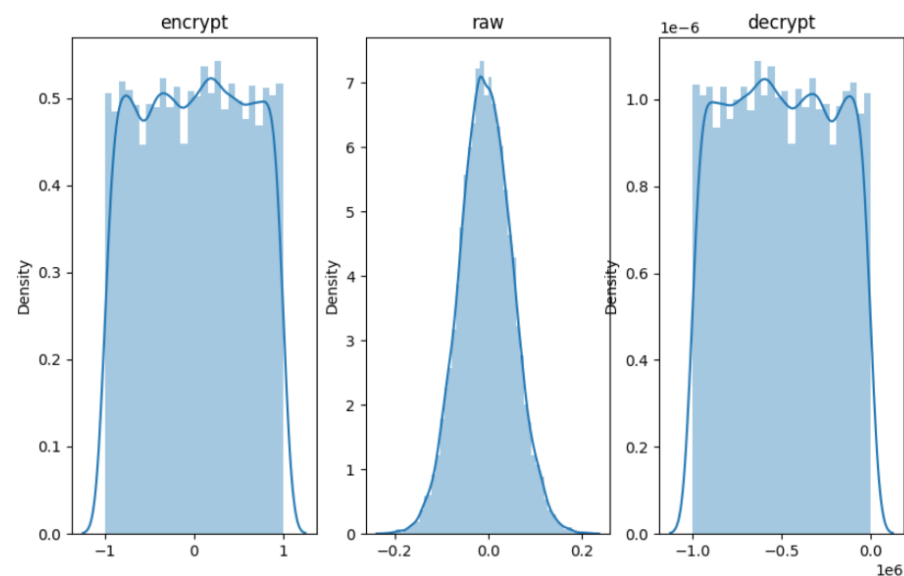
Inference Model	Without Protection	With Our Method
LSTM-CRF	82.5%	82.5%
BERT-CRF	86.2%	86.2%

From the figure, we can see that the original data distribution is a normal-like distribution with a strong statistical distribution pattern. After encryption by this method, the distribution is approximately uniform, i.e., the distribution pattern that may reveal statistical information has been masked. After decryption, the distribution can be effectively restored, and the obtained distribution information is consistent with the original

data. As shown above, the visualisation process once again proves the effectiveness of this paper's method and explains why this method does not lead to the decrease of the accuracy of the inference model. Moreover, the experimental results in the above table also demonstrate the good robustness of our method.

#### 5.4. Sensitivity Analysis

Further, to verify the data sensitivity, i.e., uncertainty, of our method, we assume that the attacker knows the encryption method in this paper and obtains the approximate key of our method by brute-force cracking, which only adds 0.1% jitter to the correct key. In this case, the data distribution visualised by cracking the encryption method is shown in Figure 6.



**Figure 6.** Data distribution of the attacker brute force at 0.1% data jitter.

From the above figure, we can see that the data obtained by brute force cracking with such a small jitter magnitude is very different from the real data distribution, which shows the high security of this method.

## 6. Conclusions

With the continuous increase and upgrade of the financial system, more and more financial entities are interconnected in numerous ways, which brings great challenges to information security because financial information will be exposed to a variety of potential threats. In this paper, we propose a privacy protection model based on cloud computing that can ensure data security even if an attacker has physical access to the cloud server. The model provides four levels of privacy protection according to users' actual needs. In the highest level of protection, the server has no access to any information about the data user, nor does it have access to the original file information of the data, nor can it restore the computational characteristics of the data, such as computational weights and gradients, and statistical characteristics of the data, such as data distribution. In addition, due to the generality of the mathematical principles of linear operator, the model can effectively protect and accelerate all models based on linear operations. The main innovations of this document are as follows:

1. The paper proposes a computing framework based on client-side encryption and decryption, accelerated computing in the cloud, and designs multiple privacy protection levels.
2. The paper adapts framework to many algorithms in the field of finance, such as Monte Carlo algorithm under linear model, n-dimensional discontinuous linear financial market model, and financial time-series analysis.



3. The paper designs a scheduling model for multi-client shared cloud GPU.
4. The paper has implemented a large number of experiments to verify the effectiveness of this method.

The final results showed that the method can increase the speed by 10 times, compared with the privacy protection method that only uses local computing power instead of the cloud server. It can also effectively prevent the user's privacy from being leaked with relatively minimal delay cost, compared with no privacy protection method. Therefore, the main advantages of this approach are fast speed, low computational loss based on privacy protection, optimised for the computational characteristics of financial scenarios, and the ability to provide different levels of protection for different usage scenarios.

Although this paper depends on the communication efficiency of the usage scenario, this limitation is somewhat alleviated considering the prevalence of fiber bandwidth and 5G high-speed networks.

In addition, although the model in this paper is tested with a large ImageNet dataset, how to deploy the model in this paper in a larger data volume environment and how to apply the model in this paper to a nonlinear computing scenario are the next research directions for the authors of this paper. At the same time, a large number of algorithms based on linear computing exist in the fields of smart agriculture, smart medicine, and remote sensing and mapping [48–56]. It will be the future work of the authors to migrate the privacy-preserving algorithms proposed in this paper to more application scenarios.

**Author Contributions:** Conceptualization, H.H. and J.G.; methodology, H.H.; software, H.H.; validation, H.H., J.G. and X.Y.; formal analysis, X.W.; writing—original draft preparation, H.H., X.W., X.Y. and J.G.; writing—review and editing, H.H., X.W., X.Y., J.G., X.L., Z.L. and J.R.; visualization, H.H.; supervision, J.R.; project administration, M.L. and J.R.; funding acquisition, J.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Nature Science Foundation of China and Beijing Social Science Foundation, grant numbers 71873129 and 19GLA002.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sato, S.; Hawkins, J.; Berentsen, A. E-finance: Recent developments and policy implications. *Tracking a Transformation: E-commerce and the Terms of Competition in Industries*; Brookings Institution Press: Washington, DC, USA, 2001; pp. 64–91.
2. Goldstein, I.; Spatt, C.S.; Ye, M. Big data in finance. *Rev. Financ. Stud.* **2021**, *34*, 3213–3225. [\[CrossRef\]](#)
3. Cockcroft, S.; Russell, M. Big data opportunities for accounting and finance practice and research. *Aust. Account. Rev.* **2018**, *28*, 323–333. [\[CrossRef\]](#)
4. Asad, M.; Ahmad, I.; Haider, S.H.; Salman, R. A critical review of Islamic and conventional banking in digital era: A case of Pakistan. *Int. J. Eng. Technol.* **2018**, *7*, 57–59. [\[CrossRef\]](#)
5. Haider, S.A.; Kayani, U.N. The impact of customer knowledge management capability on project performance—mediating role of strategic agility. *J. Knowl. Manag.* **2020**, *25*, 298–312. [\[CrossRef\]](#)
6. Bilal, Z.O.; Sulaiman, M. Factors persuading customers to adopt Islamic banks and windows of commercial banks services in Sultanate of Oman. *Rev. Int. Geogr. Educ. Online* **2021**, *11*, 651–660.
7. Du, G.; Liu, Z.; Lu, H. Application of innovative risk early warning mode under big data technology in Internet credit financial risk assessment. *J. Comput. Appl. Math.* **2021**, *386*, 113260. [\[CrossRef\]](#)
8. Cao, L. AI in finance: A review. *SSRN* **2020**, *2020*, 3647625. [\[CrossRef\]](#)
9. Matthew, K.M.; Muhammed, A.Q.; Varadarajan, V. An improved key management scheme in cloud storage. *Int. J. Adv. Intell. Paradig.* **2019**, *14*, 197–203. [\[CrossRef\]](#)
10. Quadir, M.A.; Christy Jackson, J.; Prassanna, J.; Sathyarajasekaran, K.; Kumar, K.; Sabireen, H.; Ubarhande, S.; Vijaya Kumar, V. An efficient algorithm to detect DDoS amplification attacks. *J. Intell. Fuzzy Syst.* **2020**, *39*, 8565–8572. [\[CrossRef\]](#)
11. Huttunen, J.; Jauhiainen, J.; Lehti, L.; Nylund, A.; Martikainen, M.; Lehner, O. Big data, cloud computing and data science applications in finance and accounting. *ACRN J. Financ. Risk Perspect.* **2019**, *8*, 16–30.
12. Lee, T.; Lin, Z.; Pushp, S.; Li, C.; Liu, Y.; Lee, Y.; Xu, F.; Xu, C.; Zhang, L.; Song, J. Occlumency: Privacy-preserving remote deep-learning inference using SGX. In Proceedings of the 25th Annual International Conference on Mobile Computing and Networking, Los Cabos, Mexico, 21–25 October 2019; pp. 1–17.
13. Tramer, F.; Boneh, D. Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. *arXiv* **2018**, arXiv:1806.03287.

14. Sun, Y.; Shi, Y.; Zhang, Z. Finance big data: Management, analysis, and applications. *Int. J. Electron. Commer.* **2019**, *23*, 1512270. [\[CrossRef\]](#)
15. Aydinhan, A.O.; Li, X.; Mulvey, J.M. Solving Multi-Period Financial Planning Models: Combining Monte Carlo Tree Search and Neural Networks. *arXiv* **2022**, arXiv:2202.07734.
16. Creal, D. A survey of sequential Monte Carlo methods for economics and finance. *Econom. Rev.* **2012**, *31*, 245–296. [\[CrossRef\]](#)
17. Gu, E.G.; Guo, J. BCB curves and contact bifurcations in piecewise linear discontinuous map arising in a financial market. *Int. J. Bifurc. Chaos* **2019**, *29*, 1950022. [\[CrossRef\]](#)
18. Tramontana, F.; Westerhoff, F. One-dimensional discontinuous piecewise-linear maps and the dynamics of financial markets. In *Global Analysis of Dynamic Models in Economics and Finance*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 205–227.
19. Gu, E.G. On the Price Dynamics of a Two-Dimensional Financial Market Model with Entry Levels. *Complexity* **2020**, *2020*, 3654083. [\[CrossRef\]](#)
20. Mehtab, S.; Sen, J. Analysis and forecasting of financial time series using CNN and LSTM-based deep learning models. In *Advances in Distributed Computing and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 405–423.
21. Sezer, O.B.; Gudelek, M.U.; Ozbayoglu, A.M. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Appl. Soft Comput.* **2020**, *90*, 106181. [\[CrossRef\]](#)
22. Hashemi, H.; Wang, Y.; Annaram, M. DarKnight: An accelerated framework for privacy and integrity preserving deep learning using trusted hardware. In Proceedings of the MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture, Virtual, 18–22 October 2021; pp. 212–224.
23. Zachariadis, M.; Hileman, G.; Scott, S.V. Governance and control in distributed ledgers: Understanding the challenges facing blockchain technology in financial services. *Inf. Organ.* **2019**, *29*, 105–117. [\[CrossRef\]](#)
24. Chang, V.; Baudier, P.; Zhang, H.; Xu, Q.; Zhang, J.; Arami, M. How Blockchain can impact financial services—The overview, challenges and recommendations from expert interviewees. *Technol. Forecast. Soc. Chang.* **2020**, *158*, 120166. [\[CrossRef\]](#)
25. Ertz, M.; Boily, É. The rise of the digital economy: Thoughts on blockchain technology and cryptocurrencies for the collaborative economy. *Int. J. Innov. Stud.* **2019**, *3*, 84–93. [\[CrossRef\]](#)
26. Rashideh, W. Blockchain technology framework: Current and future perspectives for the tourism industry. *Tour. Manag.* **2020**, *80*, 104125. [\[CrossRef\]](#)
27. Lin, C.; Ma, N.; Wang, X.; Chen, J. Rapido: Scaling blockchain with multi-path payment channels. *Neurocomputing* **2020**, *406*, 322–332. [\[CrossRef\]](#)
28. Janssen, M.; Weerakkody, V.; Ismagilova, E.; Sivarajah, U.; Irani, Z. A framework for analysing blockchain technology adoption: Integrating institutional, market and technical factors. *Int. J. Inf. Manag.* **2020**, *50*, 302–309. [\[CrossRef\]](#)
29. Lycklama à Nijeholt, H.; Oudejans, J.; Erkin, Z. DecReg: A framework for preventing double-financing using blockchain technology. In Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts, Abu Dhabi, United Arab Emirates, 2–6 April 2017; pp. 29–34.
30. Kar, A.K.; Navin, L. Diffusion of blockchain in insurance industry: An analysis through the review of academic and trade literature. *Telemat. Inform.* **2021**, *58*, 101532. [\[CrossRef\]](#)
31. Axon, L. Privacy-awareness in blockchain-based PKI. *Cdt Tech. Pap. Ser.* **2015**, *21*, 15.
32. Zyskind, G.; Nathan, O.; Pentland, A. Decentralizing privacy: Using blockchain to protect personal data. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 18–20 May 2015; pp. 180–184.
33. Nikolov, G.I. *Cloud Computing and Government: Background, Benefits, Risks*; Nova Science Pub.: Hauppauge, NY, USA, 2011.
34. Luftman, J.; Zadeh, H.S. Key information technology and management issues 2010–11: An international study. *J. Inf. Technol.* **2011**, *26*, 193–204. [\[CrossRef\]](#)
35. A Vouk, M. Cloud computing—issues, research and implementations. *J. Comput. Inf. Technol.* **2008**, *16*, 235–246. [\[CrossRef\]](#)
36. Sultan, N. Cloud computing for education: A new dawn? *Int. J. Inf. Manag.* **2010**, *30*, 109–116. [\[CrossRef\]](#)
37. Sarkar, P.; Young, L. Sailing the Cloud: A case study of perceptions and changing roles in an Australian university. In Proceedings of the 19th European Conference on Information Systems, ECIS 2011, Helsinki, Finland, 9–June 2011.
38. Nelson, M.R. The cloud, the crowd, and public policy. *Issues Sci. Technol.* **2009**, *25*, 71–76.
39. Denny, W.R. Survey of recent developments in the law of cloud computing and software as a service agreement. *Bus. Lawyer* **2010**, *66*, 237–242.
40. Paquette, S.; Jaeger, P.T.; Wilson, S.C. Identifying the security risks associated with governmental use of cloud computing. *Gov. Inf. Q.* **2010**, *27*, 245–253. [\[CrossRef\]](#)
41. Purushothama, B.; Amberker, B. Duplication with Trapdoor Sensitive Attribute Values: A New Approach for Privacy Preserving Data Publishing. *Procedia Technol.* **2012**, *6*, 970–977. [\[CrossRef\]](#)
42. Kamakshi, P.; Babu, A.V. Preserving the privacy and sharing the data using classification on perturbed data. *Int. J. Comput. Sci. Eng.* **2010**, *2*, 860–864.
43. Freivalds, R. Probabilistic Machines Can Use Less Running Time. In Proceedings of the IFIP Congress, Toronto, ON, Canada, 8–12 August 1977; Volume 839, p. 842.
44. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

46. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
47. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
48. Zhang, Y.; Wa, S.; Liu, Y.; Zhou, X.; Sun, P.; Ma, Q. High-Accuracy Detection of Maize Leaf Diseases CNN Based on Multi-Pathway Activation Function Module. *Remote Sens.* **2021**, *13*, 4218. [[CrossRef](#)]
49. Wang, L.; Chen, A.; Zhang, Y.; Wang, X.; Zhang, Y.; Shen, Q.; Xue, Y. AK-DL: A Shallow Neural Network Model for Diagnosing Actinic Keratosis with Better Performance than Deep Neural Networks. *Diagnostics* **2020**, *10*, 217. [[CrossRef](#)] [[PubMed](#)]
50. Zhang, Y.; Liu, X.; Wa, S.; Liu, Y.; Kang, J.; Lv, C. GenU-Net++: An Automatic Intracranial Brain Tumors Segmentation Algorithm on 3D Image Series with High Performance. *Symmetry* **2021**, *13*, 2395. [[CrossRef](#)]
51. Zhang, Y.; Wa, S.; Sun, P.; Wang, Y. Pear Defect Detection Method Based on ResNet and DCGAN. *Information* **2021**, *12*, 397. [[CrossRef](#)]
52. Zhang, Y.; He, S.; Wa, S.; Zong, Z.; Liu, Y. Using Generative Module and Pruning Inference for the Fast and Accurate Detection of Apple Flower in Natural Environments. *Information* **2021**, *12*, 495. [[CrossRef](#)]
53. Zhang, Y.; Liu, X.; Wa, S.; Chen, S.; Ma, Q. GANsformer: A Detection Network for Aerial Images with High Performance Combining Convolutional Network and Transformer. *Remote Sens.* **2022**, *14*, 923. [[CrossRef](#)]
54. Zhang, Y.; Li, M.; Ma, X.; Wu, X.; Wang, Y. High-Precision Wheat Head Detection Model Based on One-Stage Network and GAN Model. *Front. Plant Sci.* **2022**, *13*, 787852. [[CrossRef](#)] [[PubMed](#)]
55. Zhang, Y.; Wa, S.; Zhang, L.; Lv, C. Automatic Plant Disease Detection Based on Tranvolution Detection Network With GAN Modules Using Leaf Images. *Front. Plant Sci.* **2022**, *13*, 875693. [[CrossRef](#)]
56. Zhang, Y.; Wang, H.; Xu, R.; Yang, X.; Wang, Y.; Liu, Y. High-Precision Seedling Detection Model Based on Multi-Activation Layer and Depth-Separable Convolution Using Images Acquired by Drones. *Drones* **2022**, *6*, 152. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.