Research Article

Efficient Inverse Model for Cloud Computing and Information Based on the Symmetric Random Matrix

Ning Cheng and Yan Cheng

1School of Information Engineering, Hubei Light Industry Technology Institute, Wuhan, Hubei 430070, China
2Department of Computer Science and Engineering, East China University of Political Science and Law, Shanghai 201620, China

Correspondence should be addressed to Yan Cheng; chengyan@ecupl.edu.cn

Received 13 June 2022; Revised 2 August 2022; Accepted 23 August 2022; Published 21 September 2022

Copyright © 2022 Ning Cheng and Yan Cheng. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the continuous development of cloud storage technology, cloud storage has become a new way of storage for both enterprises and individuals. The cloud environment can support users to share data, but it can lead to some malicious users accessing or modifying data by illegal means, so access control is an important way to protect user data. In this paper, we propose a traceable attribute encryption scheme based on signature authentication and based on the information efficient inverse model of an asymmetric random matrix, which can trace the user who leaks information. A user attribute revocation scheme based on improved attribute encryption is also proposed, which enables data owners to reduce computational overhead. In cloud storage access control, the workload of the data owner increases exponentially when facing a high number of user attribute revocation problems, which are jointly decided by the data owner and the authorization center. An error proportional allocation (EPA) calculation method is developed to achieve an optimal estimation of system parameters in the lattice cipher scheme; experimental results show that the scheme has the advantages of shorter parameters, efficient computation, and lower storage load, proving that the scheme is resistant to IND-SAS-CPA under the standard model-based assumption of determining LWE semantic security against IND-SAS-CPA attacks; and it is used in a cloud file sharing (CFS) service framework, which can make sensitive cloud data free from the risk of privacy leakage.

1. Introduction

Attribute encryption-based access control for cloud storage is a very important research problem in cloud storage. Cloud storage can provide users with the service of sharing data, and in the era of big data, cloud storage has become a mainstream way of data storage. Therefore, the security of cloud storage is also becoming important. On the premise of being able to guarantee the security of users’ sensitive data, data owners and users also need to use the services provided in the cloud environment. Among them, access control is the most important method to secure user access to data and is the focus of this paper. Cloud computing combines a large number of resources and storage to form a larger scale shared virtual resource, enabling data resources to be shared with users through cloud computing [1]. Cloud storage is the integration of a large number of software resources, storage services, and computing services in cloud computing to form a larger-scale shared virtual resource center. Cloud storage can provide many cloud users with resources that can be expanded. It can also provide a user development environment where users in the cloud can access cloud data through client resources and can develop applications. When accessing cloud data, some sensitive data needs to be secured, and cryptography can play a large role in this. This discipline can provide more complete security for sensitive data and is currently the more critical technology in cloud storage security issues and is an important cornerstone for cloud storage to be used by many enterprises [2]. In cloud storage, data files are mainly stored in cloud servers, and data owners cannot always supervise access to data. Homomorphic encryption, especially fully homomorphic encryption, is still in the exploratory stage, and the computational complexity and efficiency are very low. The
entrusted computing scheme based on this is not very practical, and these schemes relying on homomorphic encryption are still a long way from effective implementation. After giving access to sensitive data to cloud servers, data in cloud servers can be maliciously modified or leaked at any time if a sufficiently secure access policy is not developed. Therefore, attribute encryption is particularly important in cloud storage access control, and attribute encryption is also a basic guarantee to improve the security of sensitive data. Encryption can be divided into symmetric encryption and asymmetric encryption, in symmetric encryption technology, the keys for encryption and decryption are the same and the keys are negotiated between the two users, then the security of sensitive data mainly depends on the process of key distribution [3].

Random matrix concentration inequality is an important research direction in random matrix theory, which studies the problem of limiting the probability that the extreme eigenvalues (or parametrization) of a random matrix sum are greater than a given constant and is widely used in machine learning, compressed perception, quantum computing, optimization, and other fields. As an effective tool for optimizing resource allocation, it can not only securely extend the computing power of terminals with weak computing power, such as using cloud computing platforms to pry hundreds of times more computing resources; it can also assist terminals in concentrating scattered idle computing resources to securely acquire supercomputing power, such as supercomputers in scientific computing, national defense security, and other fields. Homomorphic encryption is an ideal solution to ensure data privacy, but its immature research progress and the complexity that still needs to be improved are discouraging and not suitable for scaling; the solution using homomorphic encryption is also plagued by high computational complexity. On the other hand, there are few outsourcing computation schemes with publicly verifiable models for different user scenarios, especially for multi-user scenarios; to ensure the security of the algorithm, outsourcing can only be realized under private models, which obviously cannot meet the development trend of future user needs.

In this paper, their actual error upper bounds are studied using dimension-independent random matrix concentration inequalities, and the actual error upper bounds in the exponential form are obtained for data with very high dimensionality, and an efficient inverse model for cloud computing and information based on symmetric random matrices is constructed. From the theoretical study of sparse principal component analysis, the sparse principal component analysis is improved: to overcome the problem that its obtained principal axes are not orthogonal, the correlation of the principal axes is removed based on the idea of Schmidt orthogonalization.

2. Related Work

A random matrix is a matrix whose elements are random variables, which can also be described as random variables in the matrix space. Random matrices can be traced back to the literature [4] on the study of covariance of multivariate normal random vector samples. Liu et al. [5] considered the random matrix model with floating-point errors arising from LU decomposition in the calculation method of linear systems of equations. The literature [6] used random matrices to analyze graph theory. Zhang et al. [7] found that random matrices can be used as a model for the reaction of quantum mechanical systems. El-Yahyaoui and El Kettani [8] used the eigenvalues of random symmetric matrices to analyze the correlation of zeros of zeta functions.

Data encrypted using fully homomorphic encryption can directly perform addition and multiplication operations, which are equivalent to the corresponding operations on plaintext data, and realize the function of secure outsourcing computation, which can meet the privacy protection needs in outsourcing computation. For example, literature [9] proposes a formal definition of verifiable outsourced computation, and a general construction method for verifiable outsourced computation; literature [10] proposes a verifiable function encryption scheme and a surrogate homomorphic encryption scheme in a modular way. Thabit et al. [11] proposes a study of privacy-preserving full genetic association in the cloud environment, where all genotypic and phenotypic data are encrypted using a fully homomorphic encryption technique to protect the privacy of individuals, and the encrypted data are outsourced to the cloud using appropriate encoding and packaging techniques, and the cloud performs relevant operations on the ciphertext; the literature [12] proposes a privacy-preserving matrix decomposition algorithm and applies it to a matrix decomposition-based recommendation system, which performs matrix decomposition on the encrypted data and returns the encrypted results, so that the recommender system does not know the rating values and user summary. All these schemes are implemented based on fully homomorphic encryption techniques. Although fully homomorphic encryption schemes have evolved significantly since the Gentry blueprint, introducing many new construction techniques that have led to improvements in efficiency, Yang et al. [13] suggests that the computational complexity of fully homomorphic encryption is still too large for the average user to be applied in practice, even if a certain number of multiplication operations are sacrificed to improve the computational efficiency. In conclusion, the existing fully homomorphic encryption schemes are still unable to meet the practicality requirements due to the efficiency problem and are not promoted in outsourced computing with privacy protection. In literature [14], the concept of linear spectral statistics of the sample covariance matrix was introduced and its limit theorem was given under the satisfaction of certain conditions. Specifically, it gives a general approach to testing the overall information with sample information in a high-dimensional framework. The constructed linear spectral statistic is not only effective in solving the test problems that do not apply to statistics based on classical limit theorems but also has a high test accuracy with high generalization in a high-dimensional framework or under classical models. The literature [15] uses a parameter to define the degree of privacy leakage of data, which can ensure that
neither the addition nor the deletion of a record in the dataset affects the output result, thus making it impossible for an adversary to infer specific data from the output result. Compared with traditional privacy protection methods, differential privacy protection differs in that, it defines a strict attack model and quantifies the privacy leakage and defines the security of differential privacy based on it. Teng et al. [16] suggests that even under the premise of sacrificing a certain number of multiplication operations for the sake of operational efficiency, the computational complexity of fully homomorphic encryption is still too large for the average user and thus does not find application in practice. In conclusion, the existing fully homomorphic encryption schemes are still unable to meet the practical requirements due to the efficiency problem and are not promoted in outsourced computing with privacy protection.

3. Cloud Computing and Efficient Inverse Models Based on Symmetric Random Matrices

3.1. Differential Privacy-Preserving Techniques Based on Symmetric Random Matrices. Differential privacy protection is a privacy protection technique based on data distortion, in which noise is added to the original data to mask the data itself while keeping certain data attributes unchanged, requiring that the processed data with the noise still maintain certain characteristics and that the processed data can still be used for data analysis and data mining. Differential privacy techniques use a parameter to define the degree of privacy leakage of data, which can ensure that the addition and deletion of a record in the dataset will not affect the output results, thus making it impossible for an adversary to infer specific data from the output results. Compared with traditional privacy protection methods, differential privacy protection is different in that, it defines a strict attack model and quantifies the privacy leakage and defines the security of differential privacy on this basis. Its biggest advantage is that although differential privacy is based on data distortion techniques, the amount of noise added is independent of the size of the dataset, so for large-scale datasets, privacy protection can be achieved by adding a very small amount of noise [17]. Differential privacy protection greatly ensures the availability of data while reducing the risk of privacy leakage. Differential privacy protection is not only a general and flexible privacy protection method but also uses a strict mathematical definition to define the degree of privacy leakage, which can be used to solve many practical application problems. In particular, differential privacy techniques have been widely used in the field of statistical databases, and additionally in the application areas of data mining, machine learning, recommender systems, social networks, and genome sequencing.

The owner and the user of the outsourced data are the same users, i.e., the outsourced data is for their use only. In this case, the privacy protection of the outsourced data can be achieved by using symmetric encryption or public-key encryption, as shown in Figure 1. For example, in outsourced image processing, the image owner outsources the encrypted image to a cloud server, which implements the image processing function on the encrypted image and eventually returns the result to the user, which is an example of the self-use model in the outsourced system. Since cloud servers are not always secure, it is usually assumed that cloud servers are semi-trustworthy or malicious. Pattern matching is a fundamental problem in computer science and is a basic string operation where given a substring, the position of all substrings identical to that substring is required to be found in some string. Outsourced pattern matching is to hand over the encrypted strings to a cloud server for processing, which can be used in various applications such as biometric identification, image matching, and gene sequence query. In this paper, we propose a secure outsourced pattern matching scheme and prove its security. Compared with existing schemes, our scheme is more efficient, especially in the query phase with significant improvement.

Based on the idea of Schmidt orthogonalization, the orthogonalization of the principal axes obtained from sparse principal components is performed. After performing this modified orthogonalization, almost all the principal axes are orthogonal [18]. Compared with the original sparse principal component analysis, the improved method maintains the advantages of sparsification interpretability and removes the correlation between variables, and also allows the accuracy of the subsequent algorithm to be improved. The sparse principal component analysis is first performed to obtain the sparse loading matrix $A_t(\alpha, \beta) = F[A(t, t-1), M(i, t)]$, and then the correlations between the column vectors in the loading matrix are removed.

$$F(w) = \sum_{i=1}^{n} w(X_iw - \bar{X})^2 + |M(w)|,$$

where $M(w)$ forms the new column vector and the transformed vector is denoted as $F(w)$.

Among all the data decorrelation transformations, we want to find the transformation that projects the sample data onto the new axes as dispersed as possible, so the property to be maximized is the projection of the sample data onto the new axes. Linear transformations cannot accomplish this

![Figure 1: Privacy protection encryption diagram.](image-url)
task directly, so a property weaker than statistical independence is used: the covariance of any two variables is

$$F(\gamma) = \sqrt{a^2 + b^2 \cdot \omega + \kappa \gamma^d}. \quad (2)$$

That is, the covariance matrix of the linearly transformed data is diagonal. The diagonal covariance matrix is guaranteed to remove some redundancy but is not as effective as statistical independence.

$$F(c) = \omega \cdot [K_X + R_\eta]^{-1}. \quad (3)$$

Diagonalization of the covariance matrix also ensures statistical independence. If the range of characteristics of the data varies dramatically, then the component with the larger eigenvalue will dominate the smaller component, which will lead to biased results. Standardization can be achieved by subtracting the mean from the sample data and dividing it by the standard deviation of each component, i.e.,

$$Z(i) = \sum_{i=1}^{n} \kappa X_i + \sum_{i=1}^{n} (X_i - \bar{X})^2 + \mu_X. \quad (4)$$

The principal component analysis finds linear combinations that correspond directly to variables, i.e., principal components. It can be conducted by performing a singular value decomposition on the data or by decomposing the eigenvalues of the covariance matrix. The principal component analysis also has some drawbacks, and the main concern here is that principal components are usually linear combinations of all variables, which means that all loadings in the linear combination are usually non-zero [19]. Based on the idea of Schmitt orthogonalization, the principal axes obtained from sparse principal components are orthogonal. After performing this modified orthogonalization, almost all the principal axes are orthogonal. Compared with the original sparse principal component analysis, the improved method maintains the advantages of interpretability of sparsification and removes the correlation between variables and also allows the accuracy of subsequent algorithms to be improved.

Various correlation functions for the eigenvalues of random real symmetric matrices can be calculated by integrating over the distribution density. It is shown that the complexity of these correlation functions is strongly related to the average spacing of the eigenvalues. Differential privacy technology uses a parameter to define the degree of privacy leakage of data, which can ensure that the addition and deletion of a record in the data set will not affect the output result so that the adversary cannot infer the specific data from the output result. To simplify the results the eigenvalues are treated first. A system is in a chaotic state when the distribution of eigenvalue steps satisfies the distribution. This is because the inherent irregularities embedded in the system reach a level similar to the irregularities embedded in the random matrix. Using (3), it is possible to test whether a known system is chaotic or not [20]. If the finite time length is solved by a longer time horizon, the estimation of the covariance matrix may be affected by non-stationarity. For this reason, the covariance matrix, estimated based on historical data, will contain the effect of random "noise."

The matrix multiplication computation scheme is implemented based on two cloud servers, but this is carried out to increase the redundant computation and to greatly reduce the client’s overhead during the validation phase. The model of the system is shown in Figure 2. The model includes a client that needs computational tasks due to limited resources and two cloud servers with sufficient resources.

If cloud servers were all trusted, then clients could request cloud servers to complete computational tasks at will and would not need to guard against cloud servers stealing their private information and thus not encrypting the matrix, but actual cloud servers are often untrustworthy. In outsourced computing scenarios, almost all security threats to client data are due to improper behavior of the cloud server [21]. Outsourced computing is paid for by the client’s usage, so the server often tries to speculate about the privacy of the client’s data for financial gain. In the malicious cloud server model, the cloud server no longer strictly follows the client’s outsourcing protocol for computation; to save cost, the cloud server tries to return an incorrect result, and this incorrect result tries to pass the client’s verification of the result to save the cloud server’s computational resources. The cloud server also guesses the client’s data in an attempt to obtain private information about the client data.

The cloud server does not have access to the client’s private information. Once the client sends the data out, after the cloud server receives the data, the ownership of the data is completely in the hands of the cloud server, and the behavior of the cloud server is not controllable. So the cloud server may infer sensitive information about the client from the received data, or it may try to obtain useful information from the computation results, or even two cloud servers may conspire to steal the private information of the client. The outsourcing solution must ensure the security of the client’s data. The client can effectively detect whether the results are correct or not. Since the behavior of the cloud server is not controlled by the client, the cloud server may return some incorrect results and attempt to use the incorrect results to pass the client’s validation. The purpose of the client is to determine from the returned results whether the cloud server deviates from the protocol for the computation and successfully detect whether the returned results are the solution of the original computation task.

3.2. Efficient and Verifiable Information Efficient Inverse Model. The problem of information leakage of zero elements in matrix multiplication outsourcing is solved by adding the original matrix and a special matrix composed of vectors, and then the problem of possible linear correlation between elements is avoided by using sparse matrices. The client generates some system parameters for subsequent task encryption before outsourcing the computation task. After the cloud server completes the computation, the results returned by the cloud server are decrypted and the credentials are verified. The public delegation computation contains two features: public delegation and public verification. First, public delegation means that after the client preprocesses the computation function F and the input x, all
the information obtained can be made public and no private key is kept. Other clients can use this information directly in subsequent computations without the need to repeat the overhead preprocessing. Second, public verification means that the participating clients can verify the computation results returned by the server regardless of whether the outsourcing task is proposed or not.

In contrast, private verifiable outsourced computing schemes are relatively easy to construct as they do not require all key information to be disclosed when running the algorithm. There are currently few realistic publicly verifiable delegated computing protocols [22]. Moreover, typical protocols need to be associated with user attributes or rely on other cryptographic assumptions. Whether privately or publicly outsourced, the security of delegated computation schemes can be classified as the random oracle model and standard security model. Cryptographic mechanisms are usually based on mathematical puzzle complexity assumptions, such as factorization problems that are difficult to complete in polynomial time. Cloud computing is the most important application environment for delegated computing, and the main security issues it faces are also important challenges for delegated computing. The entrusted computing scheme for inverting a matrix is mainly based on the protocol of matrix multiplication, and the information of some types of random matrices is concentrated in a few dimensions, and there is almost no information in other dimensions. On the one hand, users delegate computational tasks to cloud servers, thus freeing them from computational resource constraints, while on the other hand, the server-side has to consume a large number of computational resources.

Taking a univariate high-order polynomial as an example, we get

$$R_f(x) = \left(\int g(t)dt \right) \cdot (x - 1) f(t). \quad (5)$$

When $t$ is very large, storing the polynomial requires a large amount of memory and the computation of the value of $f(x)$ costs a large amount of overhead. A client with limited computational power cannot afford it. Naturally, outsourcing the task of storing its already computed values is the first solution considered by researchers. To effectively protect the security of data during outsourcing, scholars often use cryptographic tools, mainly homomorphic encryption, to encrypt and decrypt to ensure data privacy. It is well known that homomorphic encryption, especially full homomorphic encryption, is still in the exploration stage and has low computational complexity and efficiency, and the practicality of delegated computation schemes based on this is not high, and these schemes relying on homomorphic encryption are still a long way from effective implementation. The introduction of signature methods is good for data security, but since the focus of this scheme is to design publicly delegable verification outsourcing schemes, how the server computes polynomial results on the encoded input is not our focus. Moreover, solutions have been proposed in the literature before, such as using encryption schemes like RSA and ElGamal to achieve homomorphic multiplication and homogeneous addition for the server to compute the encrypted data. By analyzing the existing polynomial delegation computation schemes, we found that they have problems with computational efficiency, verification properties, etc. When accessing cloud data, some sensitive data needs to be protected, and cryptography can play a greater role in it. This discipline can provide more complete security for sensitive data and is currently a key technology in cloud storage security issues. It is an important cornerstone for

**Figure 2: Dual cloud server computing model.**
cloud storage to be used by many enterprises. The basic idea of this scheme is to design the signature and verification algorithm using the abovementioned citation, which enables the client to quickly verify the computation results in a short time. It is worth noting that this topic only uses public solver keys throughout the verification process and does not involve private keys, thus achieving the goal of being publicly verifiable and delegable.

It is worth noting that we use a "shared" model to improve the efficiency of the protocol. Because the initialization phase prepares bilinear tuples, generates public-private key pairs, and encodes functions, the overhead is amortized over multiple computations [23]. Moreover, intuitively, our scheme can be easily extended to multivariate polynomial-valued delegated computation schemes, because the basic solution idea remains the same, except that the univariate in the algorithm is computed by replacing it with a multivariate. Therefore, it is also not difficult to derive its complexity analytically.

It is well known that matrix research has a long history because matrices express the association between substances in a concise and intuitive form, and operations such as inverse, transpose, and characteristic polynomial of matrices graphically express the change of association, which have important applications in real-life production. Since proposed the first secure outsourcing model, the current outsourcing scheme of matrix operations has had some relevant results, but it is generally inefficient and difficult to guarantee privacy, which obviously cannot meet the needs of the rapid development of cloud computing [24]. Given this, this chapter takes the multiplication operation in matrix operations as the research target, designs publicly verifiable delegation schemes for matrix multiplication based on the in-depth study of polynomial outsourcing computation in the previous chapters and supports dynamic update operations. More precisely, one of the matrices is labeled with a polynomial, while the other matrix is decomposed into vectors. The latter will be encoded by an algebraic one-way function on input to protect its data security, but the former data exists in plaintext and is not processed.

The delegated computational scheme for finding the inverse matrix is mainly based on the protocol of matrix multiplication, where the information of certain types of random matrices is concentrated in few dimensions, while in others there is almost no information. So far, these bounds do not explain this difference. Therefore a more precise notion of dimensionality needs to be introduced to help us distinguish between these examples, and here dimensions are introduced to help us understand this difference. Second-order optimization algorithms can obtain information about the local curvature of a surface through second-order gradients and converge faster than first-order methods in the case of high curvature. Although differential privacy is based on data distortion technology, the amount of noise added has nothing to do with the size of the dataset, so for large-scale datasets, privacy protection can be achieved by adding a very small amount of noise. Differential privacy protection greatly ensures the availability of data while reducing the risk of privacy leakage. Second-order gradient descent is usually performed by multiplying the local gradient by the inverse of the Fisher information matrix reflecting the local curvature as the update direction. The second-order optimization algorithms represented by the Newton’s method and the quasi-Newton’s method have been shown to have faster convergence than the first-order optimization algorithms in the case of quadratic functions. The convergence process is shown in Figure 3.

The curve dividing the two phases is called the critical line, and a network initialized with the point on this critical line as the sampling variance of the weights will have good expressivity. In both ordered and disordered phases, the correlation between the two signals will change exponentially, propagating forward through several layers to eventually become a fixed value that is only related to the sampling variance of the network structure and parameters, and not to the input signal to the network. In other words, initialized in the ordered and disordered phases, the correlation coefficient between signals will be forgotten exponentially; while initialized in the critical line, the correlation coefficient between signals will decay polynomially.

4. Experimental Verification and Conclusion

4.1. Factor Analysis Covariance Test. The principal component method, the sparse principal component method, and the modified sparse principal component method were used to factor analyze three sets of data and explore the results, comparing the relationship between variables and common factors, the variance contribution, the commonness of variables, and the magnitude of the difference between the covariance matrix of the original data and the decomposed covariance matrix of the three methods to analyze the similarities and differences among the three methods. The factor loading matrix reflects the degree of correlation between each common factor and each variable, and the factor loading matrix solved by the sparse principal component method directly indicates the association between each common factor and fewer features. The variance contributions of the common factors obtained by the principal component method and the sparse principal component method for the wine dataset, due to similar experimental findings, were combined with the heart data. The variance contributions of the common factors for the vehicle data were larger for the sparse principal component method compared to the principal component method because the sparse principal component method was not rotated, so the variance contributions of most of the common factors for the former were larger than those for the latter. The improved sparse principal components method did not perform as well, with the last two factors contributing much less variance to the variables.

Figure 4 shows the variable commonality obtained from the factor analysis of wine data using the three methods separately. The variable commonality indicates the dependence of the 13 original variables on the 5 common factors, and the greater the variable commonality indicates the greater the dependence on the common factors. Compared
with the principal component method, most of the variables depend more on the common factors obtained by the sparse principal component method, which indicates that the sparse principal component method loses less information than the principal component method; compared with the sparse principal component method, the improved sparse principal component method has three variables with higher commonality than the sparse principal component method, two variables with much lower commonality, and the commonality of other variables. The differences of other variables are not significant.

The performance gain of the client is defined as the ratio of the time taken by the client to complete the original task to the total time taken by the client to execute the outsourced solution, and the experimental results are analyzed next. On another hand, the original time of the client to perform the task is compared with the total time of the client to execute the outsourcing solution. The results of the comparison are shown in Figure 5. Although the difference between the outsourcing and non-outsourcing time is not significant when the matrix size is small, as the matrix increases, the non-outsourcing time grows exponentially while the non-growth of the time to execute the outsourcing scheme is very small. It shows that the modified scheme can greatly reduce the time overhead of the client. Next, the efficiency of cloud computing is analyzed. To verify the correctness of the results faster, the cloud server performs two matrix multiplication operations of the same size, so the cloud computing time should be twice as long as the original computation. However, since the computing power of the cloud server is very powerful in real applications, this part of the time is very small for the real cloud server.
The client-side performance gain is already 15 when the matrix size is 500 and increases almost linearly as the matrix size increases. Therefore, for larger matrix multiplication, the higher the client performance gain obtained by the outsourcing scheme, so the modified scheme is very suitable for outsourcing large-scale matrix multiplication. In addition, although the scheme adds redundant computations, resulting in a higher overhead in the matrix encryption phase, the client time overhead is greatly reduced in the result verification phase, so the overall efficiency of the scheme is still effectively improved.

4.2. Performance Evaluation of the Efficient Inverse Model. The theoretical analysis shows that the proposed scheme is beneficial for users to save computational overhead and storage overhead. We further use experiments to verify the results of the theoretical analysis. The personal computer (PC) configuration used in the experiments is a 3.6 GHz processor and 8 GB RAM. We chose a 1024-bit RSA alternative to the trapdoor one-way permutation in the scheme and SHA-256 instead of the fixed-length hash function H. A comparison of the computational overhead on the user side is shown in Figure 6.

Outsourcing storage and computation have arisen with the rapid development of cloud services, where resource-constrained individuals or enterprises outsource large amounts of data to cloud servers for storage and computation, which brings great benefits but also brings security and privacy leakage problems. Therefore, how to make the outsourced computation still work smoothly and get correct computation results while protecting user and data privacy has become one of the main issues in current cryptography and network security research. Cloud servers are in most cases semi-trustworthy, i.e., they can execute the protocol correctly but listen and extract secret information from users. For the same training overhead, the SBE of this paper significantly outperforms other integrated approaches in most cases. The factor loading matrix reflects the degree of correlation between each common factor and each variable, and the factor loading matrix obtained by the sparse principal component method, of the factor loading matrix directly shows the relationship between each common factor and fewer features. In particular, the SBE method in this chapter has a test error rate of 3.31% for CIFAR-10 on DenseNet-121 and 16.79% for CIFAR-100. Thus, it can be concluded that the weight scaling transformation plays a key role in increasing the diversity among network models. Moreover, the more classifications of the dataset and the larger the network model, the more significant the performance improvement.

Figure 7 shows the tested accuracy curves of various state-of-the-art deep neural network integration learning methods for training the network model PreResNet110 on CIFAR-10. The plus sign, star, and circle represent the SSE1 to SSE14 integration performance, respectively. It can be observed from Figure 7 that although the test accuracy of individual network models trained with the methods in this chapter is similar to that of other integration methods, the model integration performance obtained with the SBE method in this chapter outperforms the model integration performance obtained with other integration methods.

Test accuracy versus the number of integrated network models: in some cases, such as mobile platforms, where computational resources are limited, there is a need to balance test accuracy, test time, and the number of models involved in network model integration, so the test accuracy of network model integration obtained by running SBE versus the number of models involved in network model integration was tested. In general, the test error decreases as the number of models increases, but when the number of models reaches a certain value, the trend of decreasing the test error of model integration is weak and the test error even increases when the number of models is increased. For model ensembles to have good performance, two conditions need to be met: that is, each model has a high test accuracy; there should be high diversity among models, that is, their misclassified sample sets have a very low degree of overlap. The diversity between models can be measured by the average of the correlation coefficients between all model pairs in the model ensemble. Varying the proportion of parameters involved in the weight scaling transformation and the initial learning rate: in the algorithm, the hyperparameter $r$ ($r$ denotes the ratio of the number of weight parameters that need to perform the scaling transformation to the total number of weight parameters in the network model) has a great impact on the performance of the SBE algorithm. If $r$ is set to be small, the number of weight parameters to be scaled is relatively small and the position of the weight parameter points in the weight space changes little after the scaling transformation is performed. If $r$ is set very large, the computational overhead of training the network model until it converges again will be high.

Figure 8 shows the change curves of the test accuracy and training accuracy of the network model during the training process when different $r$ is set. As can be seen from Figure 8, after the weight scaling transformation, an equivalent
A network model is obtained theoretically, but the model test accuracy is greatly reduced by about 8% after the weight scaling transformation due to the error in the representation accuracy of the weight scaling transformation is continuously amplified in the forward propagation process, but the model training with only one epoch recovers the high test accuracy again. Although the model test error is almost the same when the model converges again for different settings of the ratio $r$, the computational overhead of training the model to converge again and the diversity of the obtained network models are different. The detection range of SBE in the weight space depends on the scaling ratio $r$. The comparison shows that the training process is more stable when the weight parameters are randomly drawn from the higher levels of the network, but after the weight scaling transformation, the weight parameters move relatively little in the weight space.

Diversity analysis of model integration: for model integration to have good performance, two conditions need to be satisfied: i.e., each model has high testing accuracy; there should be high diversity among models, i.e., the sample sets they misclassify have low overlap. The diversity among models can be measured by the average of correlation coefficients among all model pairs in the model set. The test accuracy curve for linear interpolation between two points near the same local minima is generally flat, and the test accuracy curve for linear interpolation between points near different local minima generally has a peak. The test accuracy curves of linear interpolation between network models obtained by running the Ind method have a peak, while the test accuracy curves of linear interpolation between network models obtained by running SSE and FGE are relatively flat and smooth, and the test accuracy curves of linear interpolation between network models obtained by running the SBE method show an inconsistent shape. Therefore, this thesis argues that both SSE and FGE methods try to exploit the diversity of network models around the same local minima, and Ind tries to exploit the diversity of network models under different conditions.
models with different local minima. SBE proposed in this section can be compatible with both working approaches.

5. Conclusion

Delegated computing can effectively solve the security bottleneck problem in cloud computing. The research significance of delegated computing is not only limited to this. In this paper, based on symmetric random matrices, we combine cloud computing with the establishment of an information-efficient inverse model to solve the security bottleneck problem in delegated computing and then focus on analyzing the current research status to establish the research objective of the topic as designing a publicly verifiable delegated computing scheme for polynomial and matrix operations. For the delegated computation of matrix multiplication, this paper divides one of the matrices into several column vectors, represents each vector by a polynomial and then multiplies it with the complete matrix separately and then realizes the outsourcing scheme of multiplying two matrices together. In this way, a delegated scheme for matrix multiplication can be designed based on the outsourcing computation of polynomials, which also supports dynamic update operations. Among them, we consider homomorphic hash functions to encode the undecomposed matrices to protect user data security by using their unidirectional nature; meanwhile, we adopt an attribute-based encryption regime to realize an open delegation model in a multi-user scenario. For the efficient delegation scheme of matrix inversion, this paper converts the matrix inversion operation into a function of the basic matrix operation according to the inverse matrix property of higher algebra and then extends the implemented one to construct a public verifiable delegation scheme for the inverse matrix, which fills the gap of outsourcing the inverse matrix computation under the public model. Moreover, this paper argues the correctness and security of the algorithm according to the formal definition and shows that the scheme is practicable. The efficient matrix encryption scheme proposed in this paper is applied to the recommendation system. The privacy-preserving recommendation system generally adopts matrix factorization and homomorphic encryption methods, and the solution using matrix factorization is more efficient. Leveraging our matrix encryption scheme to achieve a more efficient privacy-preserving recommender system.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the School of Information Engineering, Hubei Light Industry Technology Institute.

References

[1] M. Zhang, Y. Chen, and J. S. E.-P. P. F. M. Huang, “SE-PPFM: a searchable encryption scheme supporting privacy-preserving m in cloud systems,” IEEE Systems Journal, vol. 15, no. 2, pp. 2980–2988, 2021.
[2] A. Najafi, H. H. S. Javadi, and M. Bayat, “Efficient and dynamic verifiable multi-keyword searchable symmetric encryption with full security,” Multimedia Tools and Applications, vol. 80, no. 17, pp. 26049–26068, 2021.
[3] G. Sun and C. C. Chen, “Influence maximization algorithm based on reverse reachable set,” Mathematical Problems in Engineering, vol. 2021, Article ID 5535843, 12 pages, 2021.
[4] J. Li, X. Kuang, S. Lin, X. Ma, and Y. Tang, “Privacy preservation for machine learning training and classification based on homomorphic encryption schemes,” Information Sciences, vol. 526, pp. 166–179, 2020.
[5] Q. Liu, Y. Peng, J. Wu, T. Wang, and G. Wang, “Secure multi-keyword fuzzy searches with enhanced service quality in cloud computing,” IEEE Transactions on Network and Service Management, vol. 18, no. 2, pp. 2046–2062, 2021.
[6] A. Najafi, H. H. S. Javadi, and M. Bayat, “Verifiable ranked search over encrypted data with forward and backward privacy,” Future Generation Computer Systems, vol. 101, pp. 410–419, 2019.
[7] Y. Zhang, Y. Xiang, L. Y. Zhang, L. X. Yang, and J. Zhou, “Efficiently and securely outsourcing compressed sensing reconstruction to a cloud,” Information Sciences, vol. 496, pp. 150–160, 2019.
[8] A. El-Yahyaoui and M. D. E. C. El Kettani, “An efficiently fully homomorphic encryption scheme,” International Journal on Network Security, vol. 21, no. 1, pp. 91–99, 2019.
[9] U. Varri, S. Pasupuleti, and K. V. Kadambari, “A scoping review of searchable encryption schemes in cloud computing: taxonomy, methods, and recent developments,” The Journal of Supercomputing, vol. 76, no. 4, pp. 3013–3042, 2020.
[10] M. Tahir, M. Sardaraz, Z. Mehmoord, and M. Muhammad, “CryptoGA: a cryptosystem based on genetic algorithm for cloud data security,” Cluster Computing, vol. 24, no. 2, pp. 739–752, 2021.
[11] F. Thabit, A. P. S. Alhomydi, A. H. A. Al-Ahdal, and P. D. S. Jagtap, “A new lightweight cryptographic algorithm for enhancing data security in cloud computing,” Global Transitions Proceedings, vol. 2, no. 1, pp. 91–99, 2021.
[12] M. A. M. Ahsan, I. Ali, M. Y. I. Bin Idris, M. Imran, and M. Shoab, “Countering statistical attacks in cloud-based searchable encryption,” International Journal of Parallel Programming, vol. 48, no. 3, pp. 470–495, 2020.
[13] X. Yang, H. Zhu, F. Wang, S. Zhang, R. Lu, and H. Li, “MASK: efficient and privacy-preserving m-tree based biometric identification over cloud,” Peer-to-Peer Networking and Applications, vol. 14, no. 4, pp. 2171–2186, 2021.
[14] T. Xiao, D. Han, J. He, K. C. Li, and R. F. de Mello, “Multi-Keyword ranked search based on mapping set matching in cloud ciphertext storage system,” Connection Science, vol. 33, no. 1, pp. 95–112, 2021.
[15] A. M. Manasrah, M. A. Shannah, and M. A. Nasir, “An investigation study of privacy preserving in cloud computing environment,” Handbook of Computer Networks and Cyber Security, pp. 43–61, 2020.
[16] L. Teng, H. Li, and S. Yin, “A modified advanced encryption standard for data security,” International Journal on Network Security, vol. 22, no. 1, pp. 112–117, 2020.
[17] P. Shanthi and A. Umamakeswari, “Privacy preserving time efficient access control aware keyword search over encrypted data on cloud storage,” Wireless Personal Communications, vol. 109, no. 4, pp. 2133–2145, 2019.

[18] G. Sun, C. C. Chen, and S. Bin, “Study of cfm composite complex networks,” Symmetry, vol. 13, no. 3, p. 523, 2021.

[19] S. Guo, T. Xiang, and X. Li, “Towards efficient privacy-preserving face recognition in the cloud,” Signal Processing, vol. 164, pp. 320–328, 2019.

[20] X. Wang, X. Kuang, J. Li, J. Li, X. Chen, and Z. Liu, “Oblivious transfer for privacy-preserving in VANET’s feature matching,” IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 7, pp. 4359–4366, 2021.

[21] R. Naidu, N. Guruprasad, and V. D. Gowda, “Design and implementation of cryptcloud system for securing files in cloud,” Advances in Mathematics: Scientific Journal, vol. 9, no. 7, pp. 4485–4493, 2020.

[22] F. S. Ali, H. N. Saad, and F. H. Sarhan, “Enhance MANET usability for encrypted data retrieval from cloud computing,” Indonesian Journal of Electrical Engineering and Computer Science, vol. 18, no. 1, pp. 64–74, 2020.

[23] H. Ç. Bozduman and E. Afacan, “Simulation of a homomorphic encryption system,” Applied Mathematics and Nonlinear Sciences, vol. 5, no. 1, pp. 479–484, 2020.

[24] D. K. R. Shukla, V. K. R. Dwivedi, and M. C. Trivedi, “Encryption algorithm in cloud computing,” Materials Today Proceedings, vol. 37, pp. 1869–1875, 2021.